Exploring Engineering Applications of Visual Analytics in Virtual Reality

Sławomir Konrad Tadeja

Department of Engineering
University of Cambridge

This dissertation is submitted for the degree of
Doctor of Philosophy

Trinity Hall

September 2020

To my grandparents Stefania and Józef Wołoszyn who helped me purchase my first computer which has undoubtedly led me to this point.

Declaration

This thesis is the result of my own work and includes nothing which is the outcome of work done in collaboration except as declared in the Preface, Acknowledgements and specified in the text. I further state that no substantial part of my thesis has already been submitted, or, is being concurrently submitted for any such degree, diploma or other qualification at the University of Cambridge or any other University or similar institution except as declared in the Preface and specified in the text. It does not exceed the prescribed word limit for the relevant Degree Committee. This thesis reproduces large excerpts from my manuscripts listed in the Acknowledgements [*i*–*ix*] distributed throughout the text.

Sławomir Konrad Tadeja
September 2020

Acknowledgements

First and foremost, I would like to express my deepest gratitude towards Professor Per Ola Kristensson, who has kindly agreed to become my supervisor while I stayed at Cambridge. He has guided my research efforts throughout my entire studies and provided direction and advice whenever I found myself at the crossroads on which path should my research take. Without his help and support, I would not be able to embark and finish this remarkable adventure that working towards a PhD is.

For the majority of my stay at Cambridge, the role of my advisor belonged to Dr Patrick Langdon with whom we have had on many occasions a long and passionate discussion about the role of human factors and user-centred design in engineering. Many thanks for all this.

I would also like to express special thanks to all my colleagues from the *Engineering Design Centre*, and especial to Professor John Clarkson, who has welcomed me into his laboratory, and to Anna Walczyk and Mari Huhtala who has always found the time try and help me with all my frequent, and sometimes silly queries.

During my PhD I have had a fantastic opportunity to work alongside and collaborated with a number of scientists, many of whom become my close colleagues. I will not be able to list them all here, so I will name those with whom I collaborated the longest: Dr Pranay Seshadri, and Dr Timoleon Kipouros. Moreover, part of the research contained in this thesis concerning applications of VR into civil engineering was carried out in collaboration with Wojciech Rydlewicz, Dr Maciej Rydlewicz, and Tomasz Bubas–many thanks. Here, I would also like to thank Yupu Lu, a visitor in our laboratory with whom we have managed to draft a number of research manuscripts.

Special thanks I would like to direct towards Dr Shaona Ghosh, and Dr Jason Jacques from whom I learned tremendously, and who always found the time to answer my naive research queries and help me navigate in sometimes “turbulent waters” of my studies, and who, I would like to think, have become my friends over that time.

The friends I made whilst in Cambridge are many and the list of them is too long to be included here. I am thankful to all of them for making my life enjoyable throughout this time. Here, I would like to name a few of them in a random order, namely, Paweł Budzianowski, Maciej Tomecki, Piotr Wieprzowski, Michał Bogdan, Adam Ścibior, Marcin Machura, ks.

Waldemar Niewiński, Wojciech Rydlewicz, Karol Bacik, James Gross, Chun Yui Wong, Jonathan Mak, Bo Kang, John Dudley, Qisong Wang, Johnny O. Kim and Dr Stephen Uzor.

And I would like to acknowledge the following organisations which in one way or another supported my research: *Trinity Hall*, *Engineering and Physical Sciences Research Council (EPSRC)*, *Cambridge Commonwealth, European & International Trust*, and *Cambridge Philosophical Society*. I would also like to express my thanks to *The Alan Turing Institute* for supporting our research on the VR application in aerospace, and *Bentley Systems, Inc.* for allowing us to use in our research the photo-realistic models generated with the help of their software. I would also like to thank *Icons8 LLC* and *GrabCAD* for allowing us to use in our research projects their icons and CAD models respectively.

Finally, I would like to thank my parents Lucyna and Krzysztof Tadeja, who has loved me and supported me throughout my entire life, even when they disagreed with me. Without you, I would not be the person that I am today.

Author's Contributions

(i) **S.K. Tadeja**, P. Spurek, J.T. Jacques, P. O. Kristensson. *Practical Approaches for Providing Quality Content for VR: A Short Synthesis of Existing Methods.* (manuscript).

This article forms the basis of Section 1.4. The author's contributions are as follows: lead author; conceivement of the idea and the VR content generation index categories.

(ii) **S.K. Tadeja**, P. Seshadri, P. O. Kristensson. *Exploring Aerospace Design in Virtual Reality with Dimension Reduction.* AIAA SciTech Forum and Exposition, 7-11 Jan 2019, San Diego, CA, USA

(iii) **S.K. Tadeja**, P. Seshadri, P. O. Kristensson. *AeroVR: Immersive Visualization System for Aerospace Design and Digital Twinning in Virtual Reality.* The Aeronautical Journal, Volume 124, Issue 1280, October 2020, pp. 1615-1635

These two articles forms the basis of Chapter 3, Section 3.1. The author's contributions are as follows: lead author; interaction design; system implementation; system structure; FAST and task analyses; system verification.

(iv) **S.K. Tadeja**, Y. Lu, P. Seshadri, P. O. Kristensson. *Digital Twin Assessments in Virtual Reality: An Explorational Study with Aeroengines.* IEEE Aerospace Conference 2020. 6 Mar 2020 – 14 Mar 2020, Big Sky Conference Center, Big Sky, MT, USA, pp. 1-13

This article forms the basis of Chapter 3, Section 3.2. The author's contributions are as follows: lead author; interaction design; system partial implementation; experimental design; design implications; experiment's execution; qualitative and quantitative data analyses.

(v) **S.K. Tadeja**, Y. Lu, M. Rydlewicz, W. Rydlewicz, T. Bubas, P. O. Kristensson. *Exploring Gestural Input for Engineering Surveys of Real-life Structures in Virtual Reality using Photogrammetric 3D Models*. 2021, Multimedia Tools and Applications.

(vi) **S.K. Tadeja**, W. Rydlewicz, Y. Lu, P. O. Kristensson T. Bubas, M. Rydlewicz. *PhotoTwinVR: An Immersive System for Manipulation, Inspection and Dimension Measurements of 3D Photogrammetric Models of Real-Life Structures in VR*. (manuscript)

These two articles forms the basis of Chapter 4. The author's contributions are as follows: lead author; interaction design; system partial implementation; measurement toolkit co-design; system structure; FAST co-analysis; task analyses; experimental design; experiment's co-execution; qualitative and quantitative data co-analyses; system verification.

(vii) **S.K. Tadeja**, T. Kipouros, P. O. Kristensson. *LBW2617: Exploring Parallel Coordinates Plots in Virtual Reality*. In the Extended Abstracts of 37th ACM CHI Conference on Human Factors in Computing Systems, May 4–9, 2019, Glasgow, SCT, UK

(viii) **S.K. Tadeja**, T. Kipouros, P. O. Kristensson. *IPCP: Immersive Parallel Coordinates Plots for Engr. Design Processes*. AIAA SciTech Forum and Exposition, 6-10 Jan 2020, Orlando, FL, USA. **Aerospace Design and Structures Student Paper Competition finalist**, **AIAA Design Engineering Technical Committee Session's Best Presentation award**

These two articles forms the basis of Chapter 5. The author's contributions are as follows: lead author; interaction design; system implementation; system structure; FAST and task analyses; experimental design; design principles; experiment's execution; qualitative and quantitative data analyses; system verification and evaluation.

(ix) **S.K. Tadeja**, P. Langdon, P. O. Kristensson. *Supporting Iterative Virtual Reality Analytics Design and Evaluation by Systematic Generation of Surrogate Clustered Datasets*. IEEE International Symposium on Mixed and Augmented Reality (ISMAR 2021), 4-8 Oct, Bari, Puglia, Italy. (forthcoming)

This article forms the basis of Chapter 6. The author's contributions are as follows: lead author; interaction design; system implementation; cluster generation and colouring methods; experimental design; experiment's execution; quantitative data analyses.

Abstract

Exploring Engineering Applications of Visual Analytics in Virtual Reality
Sławomir Konrad Tadeja

Recent advancements and technological breakthroughs in the development of so-called immersive interfaces, such as *augmented* (AR), *mixed* (MR), and *virtual reality* (VR), coupled with the growing mass-market adoption of such devices has started to attract attention from academia and industry alike. Out of these technologies, VR offers the most mature option in terms of both hardware and software, as well as the best available range of different off-the-shelf offerings. VR is a term interchangeably used to denote both head-mounted displays (HMDs) and fully immersive, bespoke 3D environments which these devices transport their users to. With modern devices, developers can leverage a range of different interaction modalities, including visual, audio, and even haptic feedback, in the creation of these virtual worlds. With such a rich interaction space it is thus natural to think of VR as a well-suited environment for interactive visualisation and analytical reasoning of complex multidimensional data.

Research in *visual analytics* (VA) combines these two themes, spanning the last one and a half decades, and has revealed a number of research findings. This includes a range of new advanced and effective visualisation and analysis tools for even more complex, more noisy and larger data sets. Furthermore, the extension of this research and the use of immersive interfaces to facilitate visual analytics has spun-off a new field of research: *immersive analytics* (IA). Immersive analytics leverages the potential bestowed by immersive interfaces to aid the user in swift and effective data analysis.

Some of the most promising application domains of such immersive interfaces in the industry are various branches of engineering, including aerospace design and in civil engineering. The range of potential applications is vast and growing as new stakeholders are adopting these immersive tools. However, the use of these technologies brings its own challenges. One such difficulty is the design of appropriate interaction techniques. There is

no optimal choice, instead such a choice varies depending on available hardware, the user's prior experience, their task at hand, and the nature of the dataset.

To this end, my PhD work has focused on designing and analysing various interactive, VR-based immersive systems for engineering visual analytics. One of the key elements of such an immersive system is the selection of an adequate interaction method. In a series of both qualitative and quantitative studies, I have explored the potential of various interaction techniques that can be used to support the user in swift and effective data analysis.

Here, I have investigated the feasibility of using techniques such as hand-held controllers, gaze-tracking and hand-tracking input methods used solo or in combination in various challenging use cases and scenarios. For instance, I developed and verified the usability and effectiveness of the AeroVR system for aerospace design in VR. This research has allowed me to trim the very large design space of such systems that have been not sufficiently explored thus far. Moreover, building on top of this work, I have designed, developed, and tested a system for digital twin assessment in aerospace that coupled gaze-tracking and hand-tracking, achieved via an additional sensor attached to the front of the VR headset, with no need for the user to hold a controller. The analysis of the results obtained from a qualitative study with domain experts allowed me to distill and propose design implications when developing similar systems. Furthermore, I worked towards designing an effective VR-based visualisation of complex, multidimensional abstract datasets. Here, I developed and evaluated the immersive version of the well-known Parallel Coordinates Plots (IPCP) visualisation technique. The results of the series of qualitative user studies allowed me to obtain a list of design suggestions for IPCP, as well as provide tentative evidence that the IPCP can be an effective tool for multidimensional data analysis. Lastly, I also worked on the design, development, and verification of the system allowing its users to capture information in the context of conducting engineering surveys in VR.

Furthermore, conducting a meaningful evaluation of immersive analytics interfaces remains an open problem. It is difficult and often not feasible to use traditional A/B comparisons in controlled experiments as the aim of immersive analytics is to provide its users with new insights into their data rather than focusing on more quantifying factors. To this end, I developed a generative process for synthesising clustered datasets for VR analytics experiments that can be used in the process of interface evaluation. I further validated this approach by designing and carrying out two user studies. The statistical analysis of the gathered data revealed that this generative process for synthesising clustered datasets did indeed result in datasets that can be used in experiments without the datasets themselves being the dominant contributor of the variability between conditions.

Nomenclature

AR: Augmented Reality— **CAD**: Computer-Aided Design— **GPS**: Global Positioning System— **HMD**: Head-Mounted Display— **HPC**: High Pressure Compressor— **IA**: Immersive Analytics— **IPC**: Intermediate Pressure Compressor— **IPCP**: Immersive Parallel Coordinates Plot— **IVA**: Immersive Visual Analytics— **LPC**: Low Pressure Compressor— **IVA**: Immersive Visual Analytics— **UAV**: Unmanned Aerial Vehicle— **SDK**: Software Development Kit— **VA**: Visual Analytics— **VR**: Virtual Reality

Table of contents

Chapter 1

Introduction

Thanks to relatively recent advancements and breakthroughs in their development, so-called *immersive technologies* have begun to reach product maturity. During the second half of the last decade, we can observe a leap in the development of both the software and the hardware, concerning, in particular, the headsets (HMDs) that transport the users into bespoke, 3D environments [149]. Due to these factors, immersive interfaces are more and more frequently considered a feasible option in a wealth of applications, not only on the consumer market but also, and perhaps above all, in various industries such as the aerospace and aeronautics [157, 154, 156], computational design [150, 151], and civil engineering [153, 155].

There exist an entire spectrum of immersive technologies as shown on Milgram's reality-virtuality continuum [97]. This continuum classifies these technologies with regard to how the world is presented to the user and "how much virtual" content is used, i.e. what percentage of the content shown to the user was digitally generated (see Fig. 1.1).

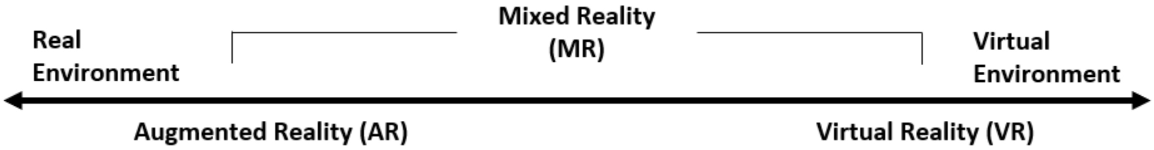


Fig. 1.1 Milgram's continuum [97]. Adapted from [122] in public domain.

Although the boundaries of terms used are rather fluid, at the moment there are two main branches of immersive technologies, namely *virtual reality* (VR) and *augmented reality* (AR). In the former case, i.e. when dealing with the VR environment, the user is completely "immersed" in the virtual world. In the context of AR, artefacts of the digitally created content are superimposed on the real surroundings either seen by the user directly through a transparent surface or composited on a smartphone or tablet screen from a camera.

There is a number of other criteria that can be used to differently classify the various branches of immersive technologies. For example, we can categorise them by (a) the method of content delivery, or (b) due to the type of device used for the content's visualisation. Either of these factors can affect the perception and effectiveness of the tasks we are planning to support with such interfaces by providing varying levels of the "immersivness", which in turn can have an impact on the user's feel of "presence" [61] or "flow" [112].

(a) Regarding the content, a video game engine can be used as a basis for building an immersive, standalone application, or the content can be presented to the user using a web-browser (WebXR[1]). The use of one of the above-mentioned methods may affect, for example, the number of people who can use our application at the same time or the quality of the displayed 3D content. Moreover, the way in which the user interacts with a given system can be substantially different in both the cases.

(b) Considering hardware, we can assign the devices into two groups depending on, for instance, the necessity of being supported by another device such as a smart-phone or a computer or if they are entirely standalone and wireless. For instance, Oculus Rift [105] must have been connected to the supporting computer via USB and HDMI cables. Whereas Oculus Quest or Go [105] HMDs were designed as wireless devices.

1.1 Virtual Reality

What currently sets *virtual reality* (VR) apart from the other branches of immersive interfaces is the current stage of its development and mass-market adoption. VR offers the most mature option in terms of both hardware and software, as well as the best available range of different off-the-shelf offerings. VR is a term is often used to denote both head-mounted displays (HMDs) and fully immersive, bespoke 3D environments that these devices transport their users to.

These headsets can be categorised depending on a range of factors, for instance, they can be supported by a computer via a cable connection (e.g. Oculus Rift [105]), or be designed as wireless devices (e.g. Oculus Quest [105]) though these devices may have less computing power built-in into the unit. This in turn, puts a number of constraints on the possible use cases for such devices. Moreover, different types of headsets offer different content rendering speed, accuracy, and quality, varying screen resolution, amount of computing power, as well as provide a range of different interaction capabilities. On the other hand, instead of an HMD, we can also use mobile devices such as tablets or smart-phones. Again, in the case of VR,

[1] https://www.w3.org/TR/webxr/

one can insert a smart-phone into a portable and cheap cardboard[2] or plastic frame[3] used to cut off the user's peripheral vision, which is necessary to achieve the effect of immersion.

In contrast to AR technology, VR has a number of challenges and limiting factors that have to be first overcome in order to unlock the full potential offered by this technology. Moreover, since the interaction paradigms are not yet understood and agreed upon by the industry and academic experts, another challenge is the selection of an appropriate interaction method. There is no optimal choice here, instead, such a choice varies depending on available hardware, the user's prior experience, the task at hand, and the nature of the input data. Furthermore, the lack of access to 3D quality-content remains an open problem.

1.2 Motivation

With modern devices, developers can leverage a range of different interaction modalities, including visual, audio, and even haptic feedback, in the creation of these virtual worlds. With such a rich interaction space it is thus natural to think of VR as a well-suited environment for (1) interactive visualisation; and (2) analytical reasoning of complex multidimensional data.

The combination of these two themes was first proposed and described in a form of an entire research agenda by Thomas et al. [161] in the second half of the first decade of the 21st century. This presentation gave birth to a new branch of research under the umbrella of *visual analytics* (VA), work in which has resulted thus far in a number of research findings. This includes a range of new advanced and effective visualisation and analysis tools for even more convoluted, noisier and larger data sets representing not only abstract, numerical data but also a wealth of more complex objects such as 3D geometric shapes or hierarchically structured, temporal data. Furthermore, the extension of this research and the use of immersive interfaces to facilitate visual analytics has spun-off yet another area of research called *immersive analytics* (IA). Immersive analytics tries to leverage the potential bestowed by immersive interfaces by enhancing data analysis drawing from the new perspectives offered by the additional dimensions and relationships exposed in the data.

Some of the most promising application domains of such immersive interfaces in the industry are various branches of engineering, including aerospace and aeronautics [157, 154, 156], computational design [150, 151], and civil engineering [153, 155]. The range of potential applications is continuously growing as new stakeholders are adopting these immersive tools.

[2] https://arvr.google.com/cardboard/

[3] https://www.samsung.com/global/galaxy/gear-vr/

1.3 Research Questions

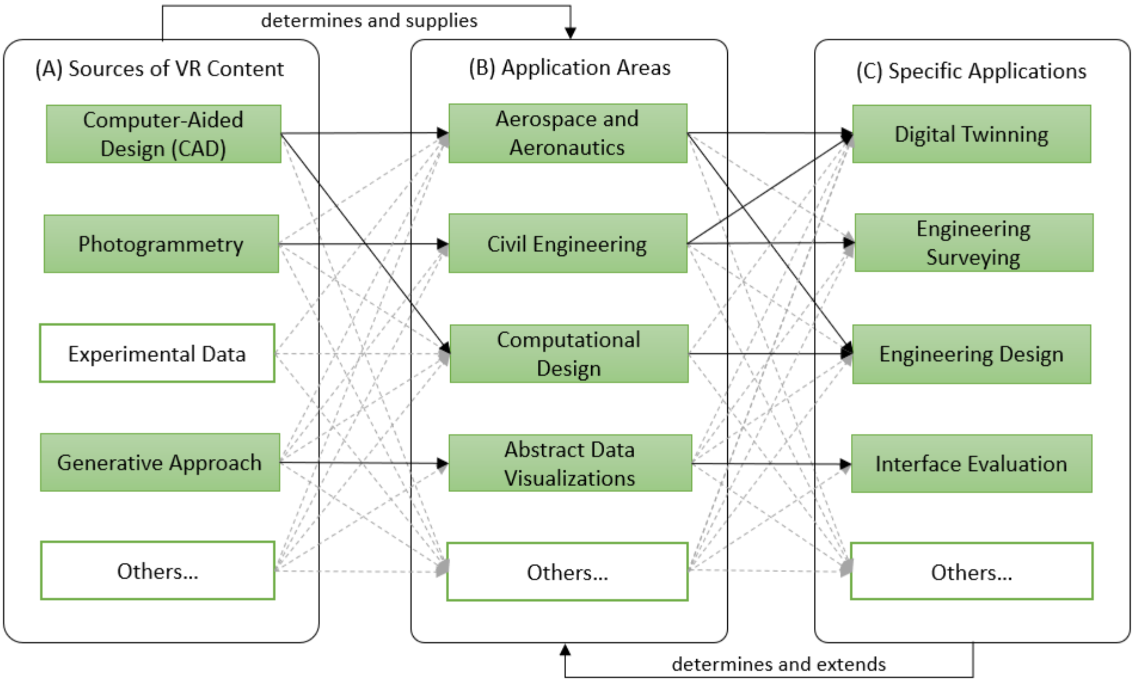


Fig. 1.2 The chart is showing the overview of the research contained in this thesis. The green boxes connected with dark arrows represent (A) a feasible source of data for particular engineering applications, or application domains (B-C) that were investigated in this thesis as a potentially promising field of adoption for VR-based analytics. Dashed arrows mark other potential avenues of research.

To this end, as the potential design space is vast and relatively unexplored, this thesis focuses on the immersive analytics supported by the VR-based environment (see Fig. 1.2 (B-C) and Fig. 1.3). However, due to its relative novelty, there still is a range of unanswered research questions (**RQ**) regarding immersive analytics and its real-world applications (see Fig. 1.3). For instance, finding an interesting use case for a VR interface can be challenging of its own. A number of potentially most promising application areas can be found in Fig. 1.2 (B-C). As mentioned before, the answers to these research questions (**RQ**) may vary greatly depending on the particular application area or a branch of the immersive technologies being used to support the research in immersive analytics. For instance, it will differ between the given engineering use cases due to the varying needs and wants of the users. This is even more apparent when designing a system for domain-experts.

When dealing with an immersive interface, one of the most important obstacles to be solved is obtaining or generating 3D content to populate these 3D virtual environments [158]. Moreover, the quality and the type of this content will have an impact regarding where and for what purpose such an immersive system can be used for (see Fig. 1.2 (A)). Having all this in mind, the first set of questions to be asked concerns providing 3D quality content for the VR-base environment as described in **RQ1**.

As, to some extent, this content simultaneously directs and limits the potential application areas, the naturally following questions relate to where and when such systems can be used. In here, this thesis identifies and discusses examples from three especially interesting application areas (see Fig. 1.2 (B) and Fig. 1.3). The first two concern mature branches of engineering where VR-based analytics can be directly applied at, namely aerospace and aeronautics [157, 154, 156], and civil engineering [153, 155]. Associated research questions can be found under **RQ2** and **RQ3** respectively.

Yet another research area that seems feasible to be tackled with the help of immersive analytics is the information visualisation in computational design within the context of aiding the decision-making processes [150, 151] (see Fig. 1.2 (B)). Such processes may require to visualise and analyse large, multidimensional datasets composed of abstract data using well-established visualisation techniques such as the Parallel Coordinates Plots [150, 151]. Related research questions can be found in **RQ4**.

Finally, **RQ5** is related to almost all immersive analytics interfaces, namely, how to evaluate such systems in a meaningful way (see Fig. 1.2 (C)). This is especially important, as visual analytics focuses on assisting the users in discovering new knowledge about the analysed data rather than on the typically used assessment procedures based on, for instance, task completion time. Here, the focus is put on the exposure of the experiment participants to the same data across all independent conditions which introduces an unavoidable learning effect.

Which such a vast design space, even partial answers to these research questions (**RQ1-RQ5**) would allow for it to be trimmed. This, in turn, would guide and provide the basis for future, more centred and focused, singular aspect studies.

1.3.1 Research Question (RQ1)

The ability to provide high-quality 3D content for VR environments is one of the key concerns when designing an immersive experience. Especially in the engineering context, the selection of content with an appropriate type and quality may determine the field of application or the particular use case. However, to a large extent, this selection task has to be carried out based on the designer's experience and prior knowledge about the available methods. In this context, based on my experience gathered during doctoral studies, at the end of this Chapter 1 will offer a short review and carry out a simple comparison exercise of the most common methods for 3D content generation.

RQ1 What considerations are required to select an appropriate content source for virtual reality?

As in any other case, there is a number of limiting factors that have to be taken under consideration when designing a given interface, whether it is going to be tools for design engineering [157, 154, 156], or an application for conducting off-line engineering surveys [153, 155]. Different types of tasks, stakeholders with different end-goals, as well as the very purpose of the given system may lead to varying sets of needs expressed by the users. These, in turn, may require different approaches when selecting an appropriate and adequate types of the 3D content that can be used in a given context and situation. To properly address this question, there are two fundamental sub-questions that have to be answered first. These questions are:

RQ1.1 What practical methods are available for providing content for virtual reality?

Currently, there are a number of techniques that allow for the generation of high-quality content for an immersive environment, including the three most practical current and promising future options i.e., (A) *computer-aided design* (CAD) supported by a number of existing, semi-automated software packages [157, 154, 156], (B) *photogrammetry* concerning reconstruction of 3D models from 2D photographs of real-world structures [153, 155], and (C) *generative approach* relying on machine learning methods for model's generation [145]. The description of these three methods including advantages and disadvantages of using each of the methods within the context of VR usage can be found in Chapter 1.4.

Moreover, Chapter 3 and Chapter 4 investigate the feasibility of using different types of 3D quality-content for the given Virtual Reality analytic tasks. The two methods of generating content for the VR environments were the 3D models using (A) computer-aided design (CAD) [157, 154, 156], and (B) photogrammetry-based techniques [153, 155], respectively.

RQ1.2 What is a meaningful way of comparing content preparation methods?

Chapter 1.4 presents carefully selected categories or characteristics derived from the analysis of the three reviewed methods (A) *computer-aided design* (CAD) [157, 154, 156], (B) *photogrammetry* [153, 155], and (C) *generative approach* [145]. Next, this framework is used in the comparison exercise that uses weighting system prescribing foreseeable importance that can be associated with each category to help decide the feasibility of employing each of the methods for a given scenario. While different characteristics may be selected, the ones used here were chosen to consider two key factors in the application of quality-content. These two factors are (1) the needs associated with a particular application area, of which usually the biggest component is the content quality requirements; and (2) the extent of resources–computer or otherwise–which are needed in order to create the content.

1.3.2 Research Question (RQ2)

RQ2 What are the benefits and obstacles in using immersive analytics in aerospace and aeronautics?

Aerospace and Aeronautics are both well-established research and industrial fields with years of development. As a consequence, the list of existing tools and currently used workflows is long and matured. For the VR-based analytics to be fully recognised and embraced by the aerospace professionals we have to first identify the areas in which an immersive interface can be successfully applied to. Due to the very nature of the VR interfaces where 3D digital models are perceived by their users as a real, tangible objects, the natural choice are the design engineering and digital twining [157, 154, 156]. These two particular application areas are further investigated under separate research questions:

RQ2.1 Can virtual reality analytics be used as a feasible alternative for design engineering in aerospace and aeronautics?

One of the potentially promising application areas of VR interface is information visualisation in aerospace [42]. In here, especially interesting is the domain of aerospace design in which solutions offering swift and effective design cycle times can be real game-changers. One of the key issue here is to provide the aerospace community with tangible results that would outlined the benefits of embracing a VR-based design framework. To address these needs, Chapter 3 introduces the *AeroVR* system for aerospace design and digital twinning. The focus of the initial, baseline system was to explore how aerospace design workflows can benefit from VR by using examples from parameter-space dimension reduction [25, 27, 131]. The lessons learned during the design, development and heuristic verification of this system with respect to usability and expressiveness are conveyed in Chapter 3. This includes function and system structures, tasks analysis, and interaction features.

RQ2.2 Can virtual reality analytics be used as a feasible tool for digital twinning in aerospace and aeronautics?

Digital twinning [35] is more and more an item of discussion in the aerospace community. Digital twins are entities mimicking characteristics of both physically existing and non-existing objects. Such twins can be used to simulate the object's behaviour and responses to given set of stimulants. With regard to this subject, Chapter 3 describes the feasibility of using the VR environment to aid and support users in conducting work related to digital twinning in aerospace and aeronautics [157, 154, 156]. The discussed immersive system is utilising VR to visualise the performance of a fleet of aircraft engines. This system uses 3D geometric computer-aided design (CAD) models of the engines paired with performance maps that characterise their nominal working condition. These maps plot pressure ratio and efficiency with superimposed the true performance of each engine. This study also explores the potential of interaction methods supported by a combination of gaze-tracking and hand-tracking achieved via an additional sensor [84] attached to the front of the VR head-mounted display, with no need for the user to hold a controller. The observational study with a small group of domain-experts helped to trim the design space and to guide further design efforts in the area of multi-object VR environment populated with digital twins.

1.3.3 Research Question (RQ3)

RQ3 Is virtual reality analytics suitable for off-line engineering surveying?

This research question concerns the feasibility of using VR-based analytics to carry out analyses over a digital twin of a given real-world object [153, 155]. In this context, we are interested in the physical characteristics of a given structure or building such as the length, surface or volume of the entire object or its individual parts. The digital twins offering such information can be bestowed by photo-realistic 3D models of real-life structures and buildings. Systems populated with such content, have the potential to improve and assist the user in performing basic off-line engineering inspection of a digitised object [153, 155]. This research question can be further split into these two sub-problems:

RQ3.1 What are the design requirements for a virtual reality analytics system for off-line engineering surveying?

Chapter 4 describes the requirements associated with the two main components of the VR system designed to conduct an engineering survey of a real-life engineering asset. These two interlinked functions are (i) 3D model manipulation, and (ii) 3D model measurement [153, 155]. The ability to manipulate the model is a significant, and potentially limiting, factor in the measurements the user may make. The viability of these design requirement was verified through an observational study with a small group of domain-experts [153, 155]. Moreover, guided by observations of participants' behaviour and their suggestions, the system was extended with new capabilities that allow the users to take surface and volume measurements of a given 3D digital object [153, 155].

RQ3.2 Is it viable to use gestural input to carry out off-line engineering surveying in virtual reality?

There is little research published to date on feasible interaction methods for VR-based systems augmented with the 3D photogrammetric models, in particular systems supporting gestural input interfaces. Consequently, Chapter 4 presents an immersive, gesture-controlled system for manipulation and inspection of 3D photogrammetric models of physical objects in VR [153, 155]. The observational study with a group of domain-expert participants validated the feasibility of the system design . This system was populated with a 3D photogrammetric models of an existing building and the domain experts were asked to carry out a survey measurement of the object using the offered measurement toolbox. The study confirmed the potential of the VR-based systems to be applied in practical real-word cases of off-line inspections of buildings and other 3D structures, and also confirms the viability of using photogrammetry as one of the promising content-delivery methods for VR as investigated for **RQ1.1**.

1.3.4 Research Question (RQ4)

RQ4 Can virtual reality analytics assist the user in exploration and analysis of a multidimensional abstract dataset in the context of computational design engineering?

Computational engineering design methods and tools are common practices in the modern industry. Such approaches are integral in enabling designers to efficiently explore larger and more complex design spaces. However, at the same time, computational engineering design methods tend to dramatically increase the number of candidate solutions that decision-makers must interpret in order to make appropriate choices within a set of solutions. Since all candidate solutions can be represented in a digital form together with their assessment criteria, evaluated according to some sort of simulation model, a natural way to explore and understand the complexities of the design problem is to visualise their multidimensional nature. The task now involves the discovery of patterns and trends within the multidimensional design space.

There is a number of well-known 2D techniques that can be used to visualise complex dataset that can be utilised in the computational design engineering. One of these methods is the well-established *Parallel Coordinates Plots* (PCP) [64]. However, translating this type of visualisation from the 2D screen onto a 3D, bespoke VR-based immersive environment is a non-trivial task [68]. The feasibility of using VR-based, IPCP visualisation requires for these two fundamental questions to be answered first:

RQ4.1 What are the design requirements for immersive parallel coordinates plots in computational design engineering?

Chapter 5 proposes the usage of the IPCP: *Immersive Parallel Coordinates Plots* [150, 151] in order to exploit and discover efficient means to use new immersive technology within a conventional decision-making process. The aim is to provide benefits by enhancing the visualisations of 3D geometry and other physical quantities with scientific information. Chapter 5 contains a description of the design of this system, which allows the representation and exploration of multidimensional scientific datasets. Furthermore, this Chapter summarises the final design decisions refined after completion of a number of experimental studies with domain experts over two separate datasets varying in both the number of data items and dimensions per data item as well. Moreover, based on the participants' observed behaviour and their comments a number of potential improvements were identified and discussed.

RQ4.2 Can immersive parallel coordinates plots assist the user in discovering new knowledge from an abstract dataset in computational design engineering?

The goal here is to enhance the design decision-making process by embedding visual analytics into an immersive virtual reality environment. To this end, Chapter 5 presents a system called IPCP: *Immersive Parallel Coordinates Plots* [150, 151]. IPCP combines the well-established parallel coordinates visualisation technique for high-dimensional data with an immersive VR interface. A qualitative evaluation with two surrogate expert users, knowledgeable in multidimensional data analysis, demonstrate that the system can be used successfully to detect both known and previously unknown patterns in a real-world test dataset, producing an early indicative validation of its suitability for decision support in engineering design processes.

1.3.5 Research Question (RQ5)

RQ5 Can artificially synthesised datasets be used to meaningfully evaluate virtual reality analytics interfaces?

It is very difficult to meaningfully evaluate the immersive analytics system in a short-term lab-based user experiment. Traditional methods, such as A/B testing, may not be the best solution here. One key factor that can specifically contribute to the complexity of A/B testing is the dataset used to conduct the user experiments.

Exposure to the same data in all independent conditions introduces a learning effect among participants. However, relying on different datasets for all experimental conditions introduces the dataset as an uncontrolled variable. To mitigate these risks, Chapter 6 presents a generative process for synthesising clustered datasets for VR analytics experiments [152]. Using this algorithm we can swiftly generate any number of distinctive datasets whilst simultaneously allowing systematic comparisons in experiments. In a two-part user study, Chapter 6 demonstrates the validity of the process and demonstrates how new insights in VR-based visual analytics can be gained using synthetic datasets generated with the help of the discussed method.

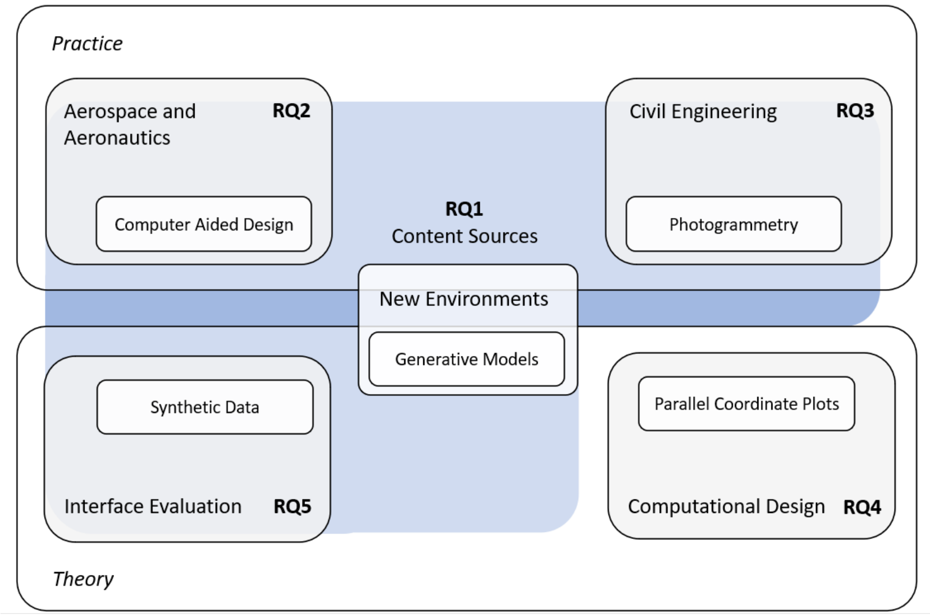


Fig. 1.3 The diagram shows how research questions (RQ) as well as the answers to these questions related to each other. In this context, the type, source, and quality of data discussed in RQ1 (blue highlighting) to a large extent underpin the RQ2, RQ3, and RQ5. The diagram also contains engineering domains under which the RQ were studied as well as provides the scope of the work carried out in this thesis.

1.4 Practical Approaches for Providing Quality Content for Virtual Reality

Over the past half-decade, research and development of technologies behind immersive 3D environments [149], such as Virtual Reality (VR), have undergone a rapid technological shift. This includes both the equipment quality as well as the way in which interaction with such devices is being carried out. VR encompasses both the virtual environment and the physical technologies required to support it, such as the HMD that allows the user to be transferred to a bespoke virtual world.

It is only sensible to assume, as seen with other technologies, that the next generations of wearable VR devices will simultaneously increase quality and reduce cost. This will, in turn, have an impact on their mass-market adoption. However, there is a range of obstacles that must be tackled in order to facilitate this process. At the current juncture there are two interlinked areas of investigation that hamper widespread adoption of VR technologies [157, 154, 155].

First, the selection of generalisable interaction techniques, appropriate and applicable to a large number of situations and environments.

Second, the ability to provide VR designers, developers, users and other stakeholders with practical and effective methods of populating these virtual worlds with quality content.

Here, I will focus on the second concern: the ability to provide high-quality content for VR environments. Using a systematic review, this Section discusses a number of methods for generating and populating VR environments with 3D content in the following three categories:

(A) **Computer-Aided Design;**
The manual or semi-automated design of 3D models using an established software.

(B) **Photogrammetry;**
The construction of 3D models from 2D photographs of real-world structures.

(C) **Generative Approach.**
The automated creation of 3D models, including machine learning-based approaches.

Each of these methods and technology types offers unique benefits to VR creators and developers. In many ways, including the availability of skills and software compatibility, long-established technologies offer a strong foundation for use in VR environments at present. However, as the use of VR expands and the needs of developers and designers grow to meet the increased demand for 3D interactive worlds, one can anticipate a shift in focus from hand-crafted 3D design to rapid generation of new and diverse environments from existing high-quality source materials.

1.4.1 (A) Computer-Aided Design

Computer-Aided Design (CAD) is commonly understood as the process of applying computers to enhance the design process. Specifically, in the context of VR, it refers to using dedicated software that aids the user in the tasks of prototyping and the creation of a particular

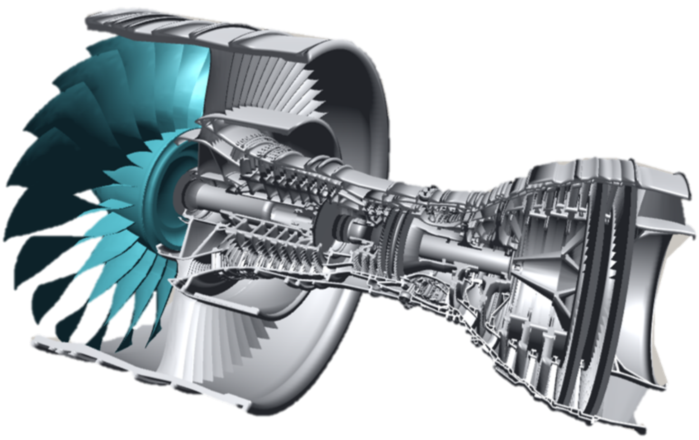


Fig. 1.4 A detailed 3D CAD model [133] of an aeroengine used as the basis for a VR-based visualisation system for aeronautics. The figure was generated within the AeroVR environment [157].

Fig. 1.5 An array of 3D CAD models [57] of aeroengines used as a basis for a VR-based digital twinning application in aeronautics [154]. The figure was generated within the AeroVR environment [157].

3D design. These tools may be packaged with Computer-Aided Engineering (CAE) software, including Finite Elements Analysis (FEA) or Computational Fluid Dynamics (CFD). CAE software packages together a set of tools that can not only help in designing an object (e.g., a product, a device, a building, a structure) but depending on its desired characteristics, can help in the simulation and testing of its potential physical properties. Equally, other tools offer the functionality of particular interest to game designers and movie makers.

While these functions may have utility in combining lighting and animation, often employed in simulations, manufacturing, computer games, and special effects, we are focused on their utility for a model generation where such output is handled by the render target for the 3D environment. In the context of providing content for VR, this can be narrowed to the Computer-Aided Geometric Design (CAGD), which Farin et al. [41] defines as: "*dealing with construction and representation of free-form curves, surfaces, or volumes.*"

The compatibility and importing of CAD-derived models into game engines, which frequently serve as frameworks used to develop VR-based applications, are an important consideration. Issues can arise due to many CAGD software packages relying on the non-uniform rational basis splines (NURBS) [41] technique to model curves and surfaces. In contrast, game engines, ordinarily use polygonal-based models (for instance, triangle meshes). This highlights the compatibility issues raised by non-trivial differences in approaches to developing models suitable for VR deployment touched on earlier. On the other hand, more and more tools are available that can assist the user in the conversion of a CAD model into a desired format.

1.4.2 (B) Photogrammetry

Photogrammetry is a practical technique [155] when highly detailed and accurate models of real-world objects are required. Photogrammetry is a method for preparing very detailed meshes of real-life objects, typically from image-based input data of the target (see Fig. 4.1). Such data can, for example, be acquired using numerous photographs taken either by hand or with the help of unmanned aerial vehicles (UAVs) for larger areas or structures [155]. Additional information, such as global positioning system (GPS) data, can be leveraged to, for instance, appropriately georeference the resulting model.

While photogrammetry can offer impressive visual models of real-world objects, this content may not always be suitable for use in VR environments. One particular concern is the completeness of the model. Where partial captures may be suitable for some applications, immersive VR experiences may require additional manipulation of the model by the designer, especially if some images or data are damaged or missing.

1.4.3 (C) Generative Approach

Generative modelling is a broad area of machine learning (ML) that models a joint distribution of data. Generative modelling can be applied to produce more examples that are similar to, but distinct from, those already present in a given dataset. For example, a generative system can start with a collection of raw images and synthesise new, unseen ones [87]. In the context of 3D modelling, such a system might take a database of 3D models of real objects and produce more similar 3D shapes to populate an environment, such as a video game [3].

One of the most promising areas of research is generating mesh representations by using only a point cloud. In a real application, the collected point cloud representation is later converted to 3D meshes by specialised software. This process can be made more effective

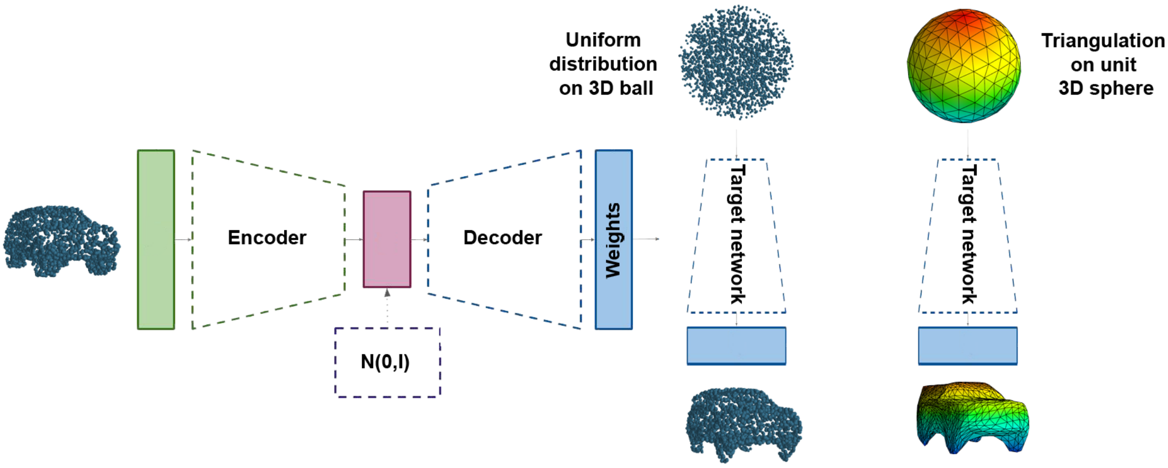

Uniform distribution on 3D ball
Triangulation on unit 3D sphere
Encoder
Decoder
Weights
Target network
Target network
N(0,I)

Fig. 1.6 HyperCloud method uses a hypernetwork architecture that takes a 3D point cloud as an input and returns the parameters of the target network. Since the parameters are generated by hypernetwork, the output dataset can be variable in size. As a result, we obtain a continuous parametrisation of the object's surface and a more powerful representation of its mesh. The diagram was adopted from Spurek et al. [145], with permission.

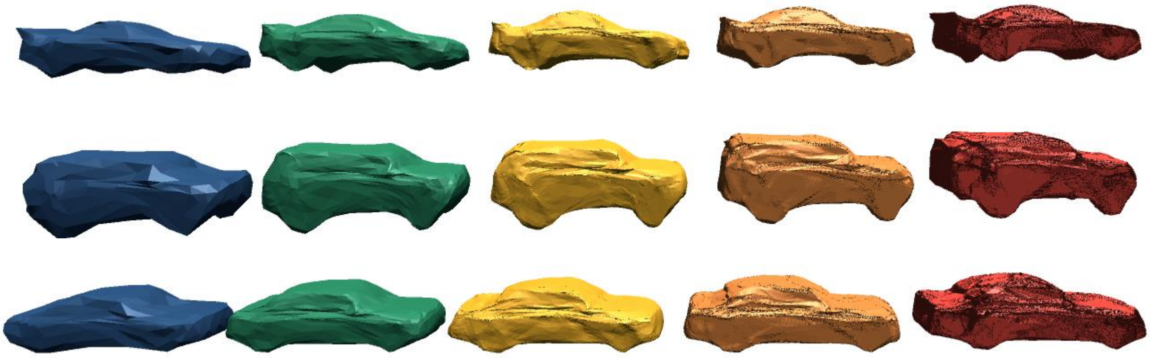


Fig. 1.7 A range of 3D meshes of cars generated with the Hypernetwork method [145] as they are seen by the headset wearer in the VR environment [157]. The number of triangles in the meshes, i.e. their quality, grows from left to right.

if VR modelling pipelines are modified to support generating meshes directly from point clouds.

One example of this technique is HyperCloud [145], a promising method that is able to generate mesh representations directly from point clouds. Instead of returning the point cloud directly, this method (based on hypernetwork approach [51]) returns the weights to the second neural network that can then be used to create an arbitrary number of points (see Fig. 1.6).

An alternative method is to use procedural generation to generate content by a combination of manual and algorithmic approaches relying on pseudo-random computing procedures.

To create fully immersive experiences, designers will be required to include substantial amounts of content which provides the setting for the actual items of interest. To avoid these backgrounds seeming unnatural it will be important to ensure there is sufficient variation. While generative modelling may excel at this, the large amount of distinctive content that could be included may have material detrimental effects on real-time rendering if not judiciously applied. While data processing remains a bottle neck, it will be important to sift and refine the generated models to ensure that the appropriate balance is struck.

1.5 Virtual Reality Content Generation Index

To assist in investigating and reflecting on the main advantages and disadvantages accompanying each of the methods for obtaining VR content, this Section proposes the VR Content Generation Index (VR CGI). The index is composed of six categories of content characteristics, which will be described next. This index provides the basis for a systematic review of different VR content generation methods. The content characteristics were selected to consider two key factors: (1) the needs associated with a particular application area, of which usually the main component is the content quality; and (2) the extent of resources required, such as computational demands, in order to create the content.

1.5.1 Content Characteristics

There are six interconnected content characteristics that influence the index. In no particular order these are:

1. **Generation Difficulty**. The difficulty of generating content affects the ability to iterate, compare and explore content options. This implies that, everything else being equal, more automated methods demanding less human input are more desirable. This is of particular concern when creating large-scale content. This content characteristic also captures the degree of control and variability available in the content generation process.

2. **Preparation Cost**. The cost and resources associated with the content generation process is another important consideration. One factor is how much manual input is required from the content creator. Depending on the quality and complexity of the model it may require less or substantially more work and, as a consequence, time of

the human modeller. Another the factor is that the user may need to use, often costly, specialised software to prepare a 3D model. A third factor is the cost of acquiring required data in order to create the content.

3. **Level of Detail**. The level of content detail can vary greatly between different content generation methods. Even within a chosen content generation method, the level of detail can vary substantially depending on how much detail is desired in the final model and how much work is required to provide more detailed content.

4. **Mimicry**. The degree content mimics real-life objects and structures is often an important consideration. The degree of mimicry can range from almost none to almost photo-realistic reconstruction of the real-life object. The photorealism of the model can be very important for certain application areas.

5. **Plasticity**. The ease of changing content is a content characteristic that captures the ability to shape or perturb the model so that it fits the needs of the designer.

6. **Adaptability and Reusability**. This content characteristic addresses the ability to adapt or reuse previously generated content for different purposes and the amount.

1.5.2 Method Comparison

The VR CGI offers a framework for reviewing methods and tools. Here, this framework is used to carry out a comparison exercise that enables distillation and reflection on the advantages and disadvantages of each approach in providing quality content for VR.

Weights will be assigned to each category. These weights capture the importance of each content characteristic, however, this importance may vary depending on the stakeholder and the use case. For example, populating a virtual forest with trees (requiring quantity, variety, low effort) demands different models than creating an archival copy of a historic monument (requiring accuracy and level-of-detail).

Three scenarios were used to illustrate the VR CGI. Scenario 1—*game scene*: this scenario illustrates applications with organic components, such as virtual forests or situations when it is desired to rapidly populate a 3D environment with varying instances of a class of objects (see Fig. 1.7). Such environments are typically found in video game scenes. In this scenario, variation in individual trees or other objects is desirable and any lower accuracy can typically be concealed in natural variations of the objects. Scenario 2—*design engineering*: The use of well-calibrated and standardised components, such as turboengines (see Fig. 1.4 and Fig. 1.5), allows simulations to be run on the composite model once the user has finished viewing and modifying it. Often such models can be based on manufacturers' CAD models of physical components. Such models typically only require limited modification before being injected into a VR environment. Scenario 3—*heritage preservation*: For digital preservation the emphasis is on highly detailed and accurate models, often necessitating redundant georeferencing methods to ensure a highly accurate final model.

Difficulty of Generation

(A) CAD is well established and there is a large CAD user base. However, in atypical situations, individual elements in the model may require manual work. This process can to some extent be supported by reusing or altering previously modelled parts or other components. However, complete automation is unavailable.

(B) Photogrammetric models are usually generated with a help of specialised apparatus and software. Once the input data is obtained it can be inserted into semi-automated photogrammetric software. Depending on available computing resources and the amount of data, the generation process can be lengthy. However, this process can be aided with the use of cloud processing tools.

The generation process will naturally rely on the actual, physical, size of the recreated model and on the desired level of detail. Thus, it is feasible to assume that, the most difficult part of the generation process is related to obtaining accurate input data at a sufficient level of detail. Examples of low-quality data may include photographs that do not have sufficient overlap or are taken with incorrect angles. This can cause difficulties for the processing algorithms to aerotriangulate the position of the cameras and recreate shapes and textures. Such inadequacies in the data may result in models that are incomplete or that lack the desired quality.

(C) A generative approach [145] has the potential to outperform in terms of generation difficulty. However, an obstacle is the necessity to obtain a sufficiently large training dataset for the neural network.

For some applications, training can be facilitated by established databases. For example, ShapeNet [20] with 55 common object categories for 3D shapes, or ModelNet [182] with 600 unique object categories. However, if there is a desire to generate specific objects that do not exist within established databases, such data must first be collected.

Once a sufficiently large training set has been established, the generation process is relatively straightforward. Once a deep neural network has been trained it can generate as many models as desired without relying on expert knowledge [145].

Preparation Cost

(A) The primary costs of content preparation in CAD is related to 1) the cost of licensing CAD software, and 2) the amount of time spent on generating models.

(B) For photogrammetry, there are three sources of content generation costs. First, licensing costs for specialised photogrammetry software. Second, acquisition costs for collecting input data, related to, among other things, the object's geographical placement and architectural complexity. Third, equipment costs, such as cameras, laser scanners and possibly drones.

(C) In the majority of instances, the model generation network can be obtained in one of the popular, free-to-use deep learning frameworks. However, to work with such packages a machine learning specialist is required to support the development process. Due to deep neural network content generation techniques being in their infancy, such freely available packages are frequently modified and updated. This in turn requires additional professional support and increases the reparation cost of the models. Nonetheless, once the technology has matured, it is expected that the user will be able to generate as many high-quality models as needed with little or no outside help. A secondary source of cost is data acquisition, if no suitable data exists to train the model on.

Level of Detail or Quality

(A) One of the main applications of CAD is in engineering where precision is critical. Thus, the level of detail offered by CAD models can be extremely high and primarily depends on the user's needs and the time spent preparing such content. The level of detail is fully controlled by the content creator and can be adapted as needed.

(B) Photogrammetry can generate 3D photo-realistic models with a very high level of detail without a content creator's direct input. However, if the model's level of detail is too high, some VR headsets may not be able to handle it effectively, hence there may be a need to reduce the level of detail. This likely requires regenerating the entire model at a lower level of detail. If a higher level of detail is required, this may require additional input data and an entirely new cycle of content creation. This dependence on data quality is why it is important to plan the acquisition phase of photogrammetry to ensure results at a desired level of detail for the target application.

(C) The use of deep learning to automatically generating 3D objects remains a recent development. It is focusing on producing new objects similar to, but not identical to, elements from a given training dataset. Consequently, new elements very often carry undesirable artefacts. For example, parts of the training model that are very close in terms of Euclidean distance can be integrated into a single mesh. In addition, the monochromaticity of generated models may make it difficult to identify some of the model's elements even though the overall 3D shape does resemble the overall object at a first glance. For practical applications, a human expert must manually correct such defects.

Mimicry

(A) CAD models can be very detailed and present ideal representations of existing objects. Moreover, the majority of products and structures are first modelled in CAD before they are manufactured. These models can be rendered to mimic how such objects appear in the real world. However, in contrast to photogrammetric models, CAD models may not necessarily accurately represent the current state of modelled objects.

(B) By its very nature, photogrammetry techniques provide up-to-date photo-realistic renderings of real world content. Some research suggests that the quality of such 3D models is sufficiently high for digitisation of heritage preservation sites [127, 6].

(C) Generative models are dedicated to producing new models that are similar to objects from a training dataset. The ability to generate new models by deep neural networks is obtained at the expense of the ability to mimic real-life objects. In its current state, the monochromaticity and level of detail of such content are not yet of sufficiently high quality for applications demanding highly detailed and accurate models.

Plasticity

(A) CAD models are generated using particular techniques (e.g., NURBS [41]) that allow simple and effective perturbation using various CAD software packages. These then can be easily transformed into a polygonal mesh format used by the majority of game engines.

(B) Photogrammetric models are usually generated semi-automatically. The manual modification requires regeneration of the entire model with either more input data, or completely new input data. This may, in turn, result in a need to acquire new data from the physical object. Typically, photogrammetric content is generated as a single continuous mesh. This mesh can be split into sub-meshes based on a selected grid. However, differentiating or itemising individual objects of a photogrammetric model require manual work or the use of specialised tools.

(C) One of the most important characteristics of generative models is the ability to generate relatively convincing samples based on interpolation or simple linear algebra. In Fig. 1.7 we show that this interpolation technique in the latent space produces smooth transitions between two distinct types of same-class objects which also demonstrates that the model is able to learn the distribution of the data correctly.

Adaptability and Reusability

(A) CAD models typically emerge as generic geometrical shapes. As various renderings are generated using a single CAD model, such content can be easily adapted in response to different requirements. It is typically relatively easy to alter the model to generate variations of existing content that can be reused in different projects.

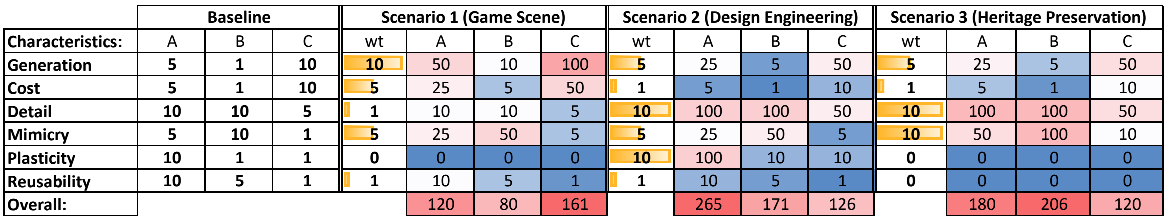
(B) Photogrammetric models have found their usage, among others, in digital twinning (e.g., [155]) and in heritage preservation (e.g., [127, 6]), as photogrammetric models reflect structures and objects in the real world. However, such photo-realistic models of a given real-life object cannot be used in another specific application that relies on information capture of a particular object's state (e.g., [155]). On the other hand, such a model can be used in training systems, or as an asset in various video games.

(C) Even though the models are in essence generated as polygonal meshes, it is difficult to alter them after generation. However, the generative method allows rapid generation of a number of models with varying levels of detail (see Fig. 1.7). Further, the deep neural network can be extended and retrained with new categories of shapes. Therefore, once the trained deep neural network has arrived at a suitable representation of the content, it provides an ability to reuse content for different applications.

1.5.3 Summary

Table 1.1 shows a summary of the content creation methods along with an example comparison of the suitability of the three content generation methods for the three different scenarios we introduced earlier in this Chapter. The six content characteristics are re-weighted for each of the three scenarios depending on their relative importance. The baseline indicates the general suitability of a content creation method. The suitability score of a particular method for a particular scenario is a linear combination of the weightings of the content characteristics and the suitability baseline score for a content creation method for a particular content characteristic. Although this method results in numerical scores, the absolute values

Table 1.1 The baseline section in the table consist of general suitability scores assigned to the content creation methods for each of the six key index content characteristics. The weightings are determined as follows. A value of 10 is high importance; 5 medium importance; 1 of little importance; 0 of no importance. The next three sections of the table show how the index can be used to assess the suitability of the content creation methods for a particular scenario. The baseline values are multiplied by the weights to pinpoint the most desired option out of the three: (A) CAD; (B) photogrammetry; and (C) a generative approach. Both the baseline values as well as the weights can be readjusted based on particular use case. The end result are linear combinations of suitability scores for each content creation method.



	Baseline			Scenario 1 (Game Scene)				Scenario 2 (Design Engineering)				Scenario 3 (Heritage Preservation)			
Characteristics:	A	B	C	wt	A	B	C	wt	A	B	C	wt	A	B	C
Generation	**5**	**1**	**10**	**10**	50	10	100	**5**	25	5	50	**5**	25	5	50
Cost	**5**	**1**	**10**	**5**	25	5	50	**1**	5	1	10	**1**	5	1	10
Detail	**10**	**10**	**5**	**1**	10	10	5	**10**	100	100	50	**10**	100	100	50
Mimicry	**5**	**10**	**1**	**5**	25	50	5	**5**	25	50	5	**10**	50	100	10
Plasticity	**10**	**1**	**1**	**0**	0	0	0	**10**	100	10	10	**0**	0	0	0
Reusability	**10**	**5**	**1**	**1**	10	5	1	**1**	10	5	1	**0**	0	0	0
Overall:					120	80	161		265	171	126		180	206	120

are not meaningful. The relative total scores can however serve as a foundation for an in-depth discussion of the advantages and disadvantages of a particular content creation method for a particular scenario.

To assist in a comparison of the suitability of VR content creation methods, this Section introduced the VR content generation index (VR CGI) and illustrated how it can be used to visualise the advantages, disadvantages, and trade-offs inherent in these content creation methods for particular application scenarios. Simultaneously, the observations regarding generation methods of content in VR tentatively answering research questions **RQ1**. As such, they also provide practitioners insight into where and with what limitations certain types of content can be used in the context of a particular engineering use case.

Chapter 2

Literature Review

Due to the highly distinctive application areas under investigation, the literature review will follow the same structure as the thesis Chapters 3–6, i.e.: applications of visual analytics in VR to (1) aerospace and aeronautics, (2) civil engineering, (3) computational design, as well as (4) evaluation of immersive visual analytics interfaces. Moreover, even though the idea of a 3D interface is decades old, only recently the technology has matured sufficiently to realise the likes of a truly immersive VR interface. Therefore, research into the application of VR with various engineering domains has only recently started to gain momentum. As such the available scientific literature on the subject is somewhat limited.

2.1 Virtual Reality Analytics in Aerospace and Aeronautics

The application of 3D interfaces, such as the one envisioned by pioneer Ivan Sutherland [149], to aid research in aeronautics and astronautics, has a decades-long history. For instance, one of the first attempts to use a VR environment for the purpose of aerospace design can be found in 1994 [52, 98].

More recently García-Hernandez et al. [42] discuss the use of VR for data mining and information visualisation, listing aerospace as a potential sector ripe for VR application. Specific applications of VR in aerospace include that of *aerodynamic design* (see also [67]) and even *spacecraft design optimisation* (see also [148]). Three areas have been previously recognised as especially promising for successful application of VR: (1) Parallel Coordinates Plots in 3D (see, for example [150] [151]); (2) complex graph visualisations; and (3) obtaining 3D data by integrating multiple 2D plots [42].

In this context, this thesis discusses two studies in aeronautics including the aerospace design in VR and assessment of aeroengines digital twins in VR outlined in Section 3.1 and Section 3.2 respectively. With respect to aerospace design, the closest in terms of goals of providing the end-users with a helpful and effective design data visualisation is the work carried out by Jeong et al. [67]. The authors used two data-mining techniques, namely analysis of variance (ANOVA) and self-organising map (SOM), however, they relied on non-immersive 2D/3D visualisation.

A myriad of other application areas at the crossroads of immersive analytics, VR and aerospace exist. This list includes, among others, collaborative environments (see [117], [23]); usage of the haptic feedback (see [124], [123]); planetary exploration (see [181]); simulation in aerospace and aeronautics (see [146]) or visualisation of sensor data and telemetry (see [181], [121], [85]).

Russel et al. [121], present an interactive VR-based visualisation for complex systems diagnosis and health management in rocket engine tests that aid non-expert users in rapidly gaining an overview of the system's operating status and health. In a similar vein, Lecakes et al. [85] discuss a VR environment for visualisation of measurements of sensory data to support the operator in prompt assessment of system health. Finally, Wright et al. [181] offer an interesting discussion of telemetry data visualisation using CAD models linked with the associated sensor data. They introduce the notion that a change in a linked data stream could result in an appropriate change in the model visualisation as well.

2.2 Virtual Reality Analytics in Civil Engineering

There are prior work examples that has investigated combining VR with photogrammetry. For example, the MOSIS (Multi-Outcrop Sharing & Interpretation System) developed by Gonzaga et al. [44] aims to leverage the benefits bestowed by an immersive environment in using photogrammetry for geoscientists working in the oil & gas industry. Similar to the system described in Chapter 4, MOSIS[44], uses the VR environment to facilitate distance measurements of photogrammetric models. It also offers several additional tools, such as a built-in compass and the ability to calculate distances between points and orthogonal distances between planes [44]. The authors [44] reflects on the fact that it is challenging to accurately capture the level of immersiveness felt by the users and therefore opted to measure the usability of the system with the System Usability Scale [44]. Liu et al. [90] discuss the benefits of integrating an Urban Geographical Information System (UGIS) with digital photogrammetry and remote sensing and couple it with interfaces such as VR to enhance urban planning and management [90]. Fritsch et al.[40] discuss the potential of using 3D photogrammetry to generate high-quality urban content for VR applications [40]. Murano et al. [99] used market-available VR hardware to simulate robotic sensor data, such as audio, video and motion sensor output, to facilitate human-robot interaction. For this type of study, the simulation must resemble as closely as possible an existing real-world environment [99]. Therefore, the authors use photogrammetry to populate their system.

While there is prior research discussing the benefits of using photogrammetry to enrich the experiences of VR users, only a few papers are specifically investigating the interactive aspects of a photogrammetry-augmented VR-based environment.

However, research on gestural input for VR in general has a long history. For example, Nishino et al. [103] introduced the *TGSH (Two-handed Gesture environment SHell)* interface.Song et al. [143] discussed the *Finger-Gesture* method for selection and manipulation in VR, while Latoschik [83] presented a VR designated framework for gesture detection and analysis.

In contrast, few have specifically investigated gestural input as the main interaction method for systems coupling VR with photogrammetry. One example is See et al. [127], which studied the usability of virtual hands facilitated by hand-held controllers to increase the feeling of presence in the VR environment. In their study, the authors [127] compared two conditions in which the participants were either shown virtual hands or avatars of the controllers [127]. The study concluded that, overall, using even low fidelity virtual hands is more favourable. Zhang et al. [185] proposed to augment VR controllers with an attached, near-infrared camera to track and display the hand and finger postures. Juckette et al. [70] mention the use of a Leap Motion device [84] to facilitate the gesture-interface for the VR experience for exploration of an ancient Maya temple [70]. Manders et al. [92] proposed a technique that leverages the fact that HMDs leave the user's chin visible for tracking. Coupled with stereo camera data to capture the user's gestural input, the user's chin can be used to manipulate 3D objects in virtual space.

Another interesting area of research is how the appearance and fidelity of hand avatars in VR affect the user. In addition to the previously discussed studies [127, 185], Lin et al. [89] investigated how different virtual hand models impact the user's body ownership illusion (BOI). The authors [89] reflected, that, although the results were inconclusive, using a more realistic human-hands model might be favourable. Tecchia et al. [159] describes a system that uses colour-coded thimbles (for thumb and index fingers) tracked by an RGBD camera (i.e. RGB depth camera) attached to the headset for real-time projection of the user's hands and body in the Mixed Reality (MR) environment. They hypothesise that such realistic hand-models can strongly impact the user's sense of presence [159].

2.3 Virtual Reality Analytics in Computational Design

Since the PCP concept was initially proposed by Inselberg [62] the number of entries containing the phrase “parallel coordinates”, as returned by Google Scholar search on July 12$^{\text{th}}$ 2021, reaches over two‘ million items. The theory and applications of PCP can be found in the textbook written by Inselberg [63].

The wide adoption of 2D PCP has led to multiple attempts of designing a useful and compelling 3D PCP. For example, Wegenkittl et al. [173] and Gröller et al. [48] describe Extruded PCP, which uses a PCP to visualise a complex surface. A different example of using the surface-type design is a density isosurface [147], which was used to visualise cytomics data. Falkman [37] describes the *Cube* package applied for swift analysis of medical data.

Parallel Glyphs combine Star Glyphs [113] with 3D PCP [38]. The usability of 3D Multi-Relational PCP in conjunction with High-Precision Textures has been investigated in Johansson et al. [69]. Dang et al. [33] describe a few examples of 3D parallel dot displays and stacked PCP, which uses various stacking techniques to augment the standard visualisation of dot plots and PCP respectively. In their related work, Chang et al. [21] compared joined side-by-side 2D visualisations to PCP with scatter plot matrices, concluding that the PCP-based approach offered clear advantages over individual visualisations for most of their participants for a number of their tasks. Furthermore, Johansson et al. [68] presented a comparison between 3D PCP visualisations with conventional 2D plots and concluded in favour of the latter. Holten et al. [59] discuss different variants and improvements of 2D PCP used to find clusters in the multivariate datasets and remarks that only the use of additional scatter plots has a substantial positive impact on the visualisation.

ImAxes [28] allows the user to rearrange the axes interactively and supports both 2D and 3D scatter plots. Another example of an immersive PCP is the ART collaborative data analysis tool [15] that distributes the user interface across an augmented reality headset and a touch surface. Rosenbaum et al. [119] took another approach in which the user was treated as an element of the immersive PCP visualisation itself.

The approach to design immersive PCP (IPCP) described first discussed in Tadeja et al. [150] and later on discussed in Tadeja et al. [151], differs from other attempts in several key ways. First, the IPCP denote values in each dimension with interactive volumetric objects (i.e., unit-size cubes) that automatically highlight the item under the user’s gaze in order to signal that they can be interactive. The same is true for all the other interactive elements, such as selectors (i.e., various size spherical markers in the interface) used to move or rotate the scatter plots. This in turn opens up new design possibilities [114] that can use characteristics of a 3D shape, such as its orientation in the 3D space [114]. In addition, users can freely move within and manoeuvre in the “infinite” 3D space to explore all parts of the visualisation

[150]. Furthermore, the IPCP system augments the pattern identification task with 3D scatter plots [156, 150, 157, 151], that can substantially aid users in isolating groups of data items that can potentially form patterns.

A previous study conducted by Johansson et al. [68] indicated that 3D PCP was, in fact, worse than a conventional 2D version in both the metrics (i.e., response times and error rates). However, the study was conducted using a desktop computer connected to an external display instead of an immersive, VR environment, and thus it is unclear whether the findings generalise to a truly immersive environment. However, such a study is intrinsically limited in terms of the many study parameters that have to be arbitrarily set in order to be able to achieve an A/B comparison. Examples of such study parameters include data density and volume, saliency of data items, field of view, the user's sense of presence, situational awareness of a 3D world etc.

2.4 Evaluation of Virtual Reality Analytics Interfaces

Evaluating visualisations in general is acknowledged to be challenging [177]. A systematic overview of these issues can be found in Isenberg et al. [64]. In general, the 3D world data category identified by Shneiderman et al. [138] is a visualisation category which *"is still controversial"* [138] and brings additional challenges, echoed by many others [136, 42, 13, 80].

Comparing different three-dimensional information visualisation designs is a difficult problem, as pointed out by for example Wiss et al. [180]. Certain designs are not suitable for some datasets and not every interface can support all the fundamental user tasks as listed by Shneiderman [135, 180]. Using artificial data as part of the experimental design is not unheard of [59, 78, 9, 184]. For instance, Holten et al. [59] synthesised a number of cluster to evaluate different design variants of Parallel Coordinate Plots (PCP) [151], which is a widely used technique for multivariate data visualisation and cluster identification. Holten et al. [59] used Gaussian distributions to generate the point clouds and background noise [59]. Generative approaches using Gaussian distributions have also been proposed by Sedlmair et al. [126], Bach et al. [9], and Prouzeau et al. [109].

Clusters represent an important class of features [78, 9, 126]. However, they are not the only ones encountered in real-world data [94, 93]. For example, the *scaganostics* approach that describes various methods of interpreting appearance of data on the scatterplots or graphs [164] catalogues and reasons about data traits, such as trends, outliers, smears or other possible anomalies [164, 178, 179]. Further examples of synthesizing datasets with various characteristics can be found in Theodoridis et al. [160], Matejka et al. [94] and Manniono et al. [93].

Another approach to data generation was proposed by Yang et al. [184], which used the MINST dataset of handwritten digits [86] as well as the t-SNE dimensionality reduction technique [168] to synthesise point clouds. Filho et al. [39, 171] relied in their experiments on a dataset constructed out of real-world voting data. Furthermore, Bach et al. [10] proposed a method of generating random graph data.

Chapter 3

Virtual Reality Analytics in Aerospace and Aeronautics

One of today's most propitious immersive technologies is *virtual reality* (VR). This term is colloquially associated with headsets that transport users to a bespoke, built-for-purpose immersive 3D virtual environment. It has given rise to the field of immersive analytics—a new field of research that aims to use immersive technologies for enhancing and empowering data analytics. However, in developing such a new set of tools, one has to ask whether the move from standard hardware setup to a fully immersive 3D environment is justified—both in terms of efficiency and development costs. To this end, this Chapter will discuss two independent case studies concerning the feasibility of using VR for (a) aerospace design, and (b) digital twinning, presented in the next two Sections 3.1-3.2, whereas Section 3.3 contains the final remarks.

3.1 Aerospace Design in Virtual Reality

Virtual reality (VR) is rapidly being hailed as the new paradigm for interactive visualisation of data. Its ability to fuse visual, audio, and haptic sensory feedback in a computer-generated simulation environment is deemed to have tremendous potential. While the phrase *virtual reality* has been used for decades, in the context of computer-aided visualisation, today it is synonymous with head-mounted displays [149] (HMDs) or headsets [105, 104]. Although still in a nascent stage, HMDs have demonstrated their usefulness in the computer gaming, education, fashion and real-estate industries, with countless more application areas currently being pursued, including information visualisation in aerospace [42]. One potentially promising application is aerospace design—a complex, multi-disciplinary, multi-objective and multi-dimensional problem—where technologies that offer faster design cycle times, with potentially greater efficiency gains, can be real game-changers. However, as the aerospace community usually works on state-of-the-art computational tools and sophisticated computer-aided design packages, there are tremendous hurdles in getting the community to embrace VR. Furthermore, at this stage, it is not precisely clear what the benefits are in migrating to a VR-based design framework. Thus, what is required by the aerospace design community is an initial sketch of an immersive VR aerospace design environment—the AeroVR, a computer-generated environment that leverages the full visual, audio, and haptic sensory frameworks afforded by VR technology.

The focus here is to explore how aerospace design workflows can benefit from VR. To aid our effort, we will be using ideas from parameter-space dimension reduction [25, 27, 131]. This topic has recently received considerable attention from both the applied mathematics and computational engineering communities, where the aim has been to reduce the cost of expensive computer parameter studies—that is, optimisation, uncertainty quantification, and more generally *design of experiments*. In Sub-Section 3.1.1, we present some of the key theoretical ideas that underpin dimension reduction. This is followed in Sub-Sections 3.1.2–3.1.9 with a presentation of the VR aerospace design environment including its function and system structures, tasks analysis, and interaction features. The next Sub-Section 3.1.10 describes a verification of the interface with respect to usability and expressiveness.

This work represents one of the first forays of virtual reality in aerodynamic design—a field where state-of-the-art computer aided design (CAD) tools are quite competent. Given the nascent stage of VR technology, the objective of this research is to gauge its potential and lay down some of the key building blocks for future digital twinning and aerospace design efforts. In the years to come, the AeroVR concept will have to be carefully evaluated and compared with existing CAD tools through, for example, a series of controlled experiments to demonstrate superiority in executing design tasks. This latter task is beyond the scope of this thesis and will require years of collaborative research involving industry and academia. Our work here is the first step in that direction.

3.1.1 Parameter-Space Dimension Reduction

Consider a function $f(\mathbf{x})$ where $f : \mathbb{R}^d \to \mathbb{R}$. Here f represents our chosen quantity of interest (qoi); the desired output of a computational model. This qoi can be the lift coefficient of a wing or indeed the efficiency of a turbomachinery blade. Let $\mathbf{x} \in \mathbb{R}^d$ be a vector of design parameters. Now when $d \leq 2$, visualising the design space of f is trivial, one needs to simply run a design of experiment and view the results as a scatter plot. However, when $d \geq 3$

visualising the design space becomes difficult. One way forward is to approximate f, with

$$f(\mathbf{x}) \approx g(U^T \mathbf{x}), \tag{3.1}$$

where $g : \mathbb{R}^m \rightarrow \mathbb{R}$ and $U \in \mathbb{R}^{d \times m}$, with $m \leq d$. We call the subspace associated with the span of U its *ridge subspace* and $g(U^T \mathbf{x})$ its *ridge approximation*. Further, we assume that the columns of U are orthonormal, i.e., $U^T U = I$. The above definitions imply that the gradient of f is nearly zero along directions that are orthogonal to the subspace of U. In other words, if we replace $\mathbf{x}$ with $\mathbf{x} + \mathbf{h}$ where $U^T \mathbf{h} = 0$, then $f(\mathbf{x} + \mathbf{h}) = g(U^T(\mathbf{x} + \mathbf{h})) = f(\mathbf{x})$. visualising f_i along the coordinates of $U^T \mathbf{x}_i$ for all designs i, can provide extremely powerful inference; such scatter plots are called *sufficient summary plots*. To clarify, let $\mathbf{u}_i = U^T \mathbf{x}_i$ where $\mathbf{u} \in \mathbb{R}^m$. Let us assume that $m = 1$, in which case we can collapse all the $\mathbf{x}_i$ to $\mathbf{u}_i$ via U into a 1D scatter plot. Here $\mathbf{u}_i$ values would lie along the horizontal axis, while f_i values would be plotted along the vertical axis. Such a sufficient summary plot can be useful for characterizing and understanding the relationship between f and $\mathbf{x}$. We offer practical examples of this recipe in the forthcoming section.

Techniques for Dimension Reduction

Techniques for estimating U build on ideas from sufficient dimension reduction [27] and more recent works such active subspaces [25] and polynomial [58, 26] and Gaussian [132] ridge approximations. While our work in this Section is invariant to the specific parameter-space dimension reduction technique utilised, we briefly detail a few ideas within ridge approximation. Our high-level objective is to solve the optimisation problem

$$\underset{U \in \mathbb{R}^{d \times m}, \, \alpha \in \mathbb{R}^p}{minimise} \left\| f(\mathbf{x}) - g_{\alpha}\left(U^T \mathbf{x}\right) \right\|_2^2, \tag{3.2}$$

over the the space of matrix manifolds U and the coefficients (or hyperparameters) α associated with the parametric function g. This is a challenging optimisation problem and it is not convex. In Seshadri et al. [132] the authors assume that g is the posterior mean of a Gaussian process (GP) and iteratively solve for the hyperparameters associated with the GP, whilst optimizing U using a conjugate gradient optimizer on the Stiefel manifold (see Absil et al. [2]). In Constantine et al. [26] the authors set g to be a polynomial and iteratively solve for its coefficients—using standard least squares regression—whilst optimizing over the Grassman manifold to estimate the subspace U. It should be noted that these techniques are motivated by the need to break the *curse of dimensionality*. In other words, one would like to estimate both g and U for a d dimensional, scalar-valued, function f without requiring a large number of computational simulations.

The dimension reduction strategy we pursue in this Section is based on active subspaces [25] computational heuristic tailored for identifying subspaces that can be used for the approximation in (3.2). Broadly speaking, active subspaces requires the approximation of a covariance matrix $C \in \mathbb{R}^{d \times d}$

$$C = \int_{\mathcal{X}} \nabla_x f(\mathbf{x}) \nabla_x f(\mathbf{x})^T \rho(\mathbf{x})\, d\mathbf{x}, \tag{3.3}$$

where $\nabla f(\mathbf{x})$ represents the gradient of the function f and ρ is the probability density function that characterizes the input parameter space $\mathcal{X} \in \mathbb{R}^d$. The matrix C is symmetric positive semi-definite and as a result it admits the eigenvalue decomposition

$$C = W \Lambda W^T = \begin{pmatrix} W_1 & W_2 \end{pmatrix} \begin{pmatrix} \Lambda_1 & \\ & \Lambda_2 \end{pmatrix} \begin{pmatrix} W_1 & W_2 \end{pmatrix}^T, \tag{3.4}$$

where the first m eigenvectors $W_1 \in \mathbb{R}^{d \times m}$, where $m << d$—selected based on the decay of the eigenvalues Λ—are on average directions along which the function varies more, compared to the directions given by the remaining $(d-m)$ eigenvectors W_2. Readers will note that the notion of computing eigenvalues and eigenvectors of an assembled covariance matrix is analogous to principal components analysis (PCA). However, in (3.3) our covariance matrix is based on the average outer product of the gradient, while in PCA it is simply the average outer product of samples, i.e., $\mathbf{x}\mathbf{x}^T$. Now, once the subspace W_1 has be identified, one can approximate f via

$$\begin{aligned} f(\mathbf{x}) &= f\left(W W^T \mathbf{x}\right) = f\left(W_1 W_1^T \mathbf{x} + W_2 W_2^T \mathbf{x}\right) \\ &\approx g\left(W_1^T \mathbf{x}\right), \end{aligned} \tag{3.5}$$

in other words we project individual samples $\mathbf{x}_i$ onto the subspace W_1. Moreover, as the function (on average) is relatively flat along directions W_2, we can approximate f using the directions encoded in W_1.

But how do we compute (3.3), as for a given f we may not necessarily have access to its gradients? In [129], the authors construct a global quadratic model to a 3D Reynolds Averaged Navier Stokes (RANS) simulation of a turbomachinery blade and then analytically estimate its gradients. We detail their strategy below as we adopt the same technique for facilitating parameter-space dimension reduction.

Assume we have N input-output pairs $\{\mathbf{x}_i, f_i\}_{i=1}^{N}$ obtained by running a suitable design of experiment (see [111]) within our parameter space. We assume that the samples $\mathbf{x}_i \in \mathbb{R}^d$ are independent and identically distributed and that they admit a joint distribution given by $\rho(\mathbf{x})$. Here we will assume that $\rho(\mathbf{x})$ is uniform over the hypercube $\mathcal{X} \in [-1,1]^d$. We fit a global quadratic model to the data,

$$f(\mathbf{x}) \approx \frac{1}{2}\mathbf{x}^T \mathbf{A}\mathbf{x} + \mathbf{c}^T \mathbf{x} + d, \tag{3.6}$$

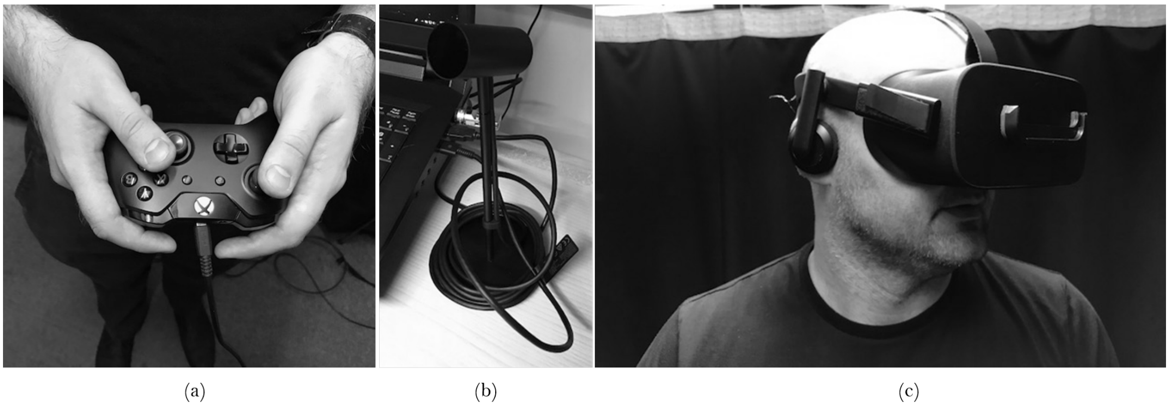
using least squares. This yields us values for the coefficients $A, \mathbf{c}$ and the constant d. Then, we estimate the covariance matrix in (3.3) using

$$\hat{C} = \int_{X} (A\mathbf{x} + \mathbf{c})(A\mathbf{x} + \mathbf{c})^T \rho(\mathbf{x})\, d\mathbf{x}. \tag{3.7}$$

Following the computation of the eigenvectors of $\hat{C}$, one can then generate sufficient summary plots that are useful for subsequent inference and approximation.

We apply this quadratic recipe and show the sufficient summary plots for a 3D turbomachinery blade in Section 3.1.2, both in a standard desktop environment and in virtual reality. This comparison—the central objective of this research—is motivated by the need to explore the gains VR technologies can afford in aerospace design. That said, prior to delving into our chosen case study, an overview of existing immersive visual technologies and their associated frameworks is in order.

3.1.2 Supporting Aerospace Design in VR



(a) (b) (c)

Fig. 3.1 Hardware interfaces in VR: (a) Xbox controller; (b) Oculus Rift's motion sensor; (c) Oculus Rift VR headset.

Visual analytics (VA), a phrase first coined by J.J. Thomas et al. [161] has two ingredients: (1) an interactive visual interface [161]; and (2) analytical reasoning [161]. Recent advances and breakthroughs in the development of VR (virtual reality) and AR (augmented reality) have spun off another branch of research known as *Immersive Analytics* [19] (IA). IA seeks to understand how the latest wave of immersive technologies can be leveraged to create more compelling, more intuitive and more effective visual analytics frameworks. There are still numerous hurdles to overcome for VR-based data analytics tools to be widespread. Issues associated with any type of a 3D interface, for example, potential occlusion effects [137], high computing power demands and specialised, (often costly) hardware, have to be resolved, or at least minimised. Moreover, interaction techniques have to facilitate a user's understanding of the visualisation and avoid becoming a distraction. Finally, certain guidelines have to be incorporated to mitigate the risk of the simulation sickness [73, 120] symptoms that can manifest during or immediately after the use of a VR headset.

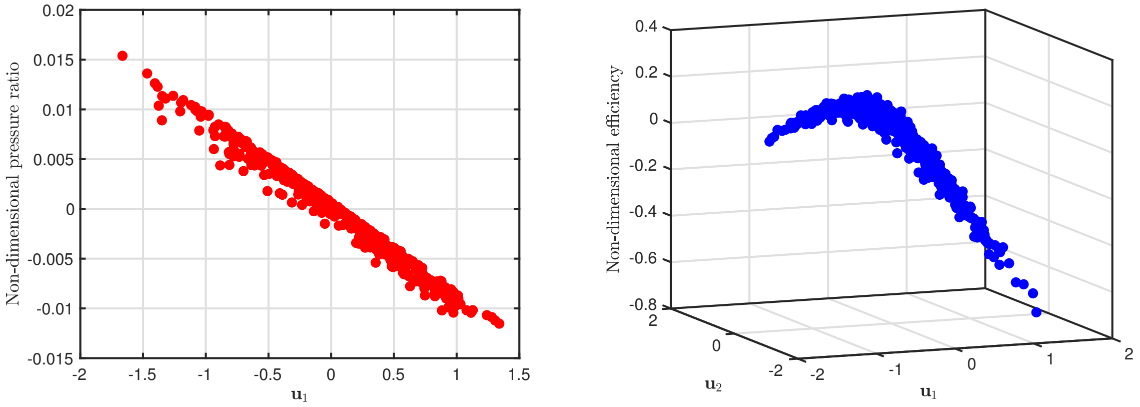
Many interaction techniques and devices have been conceived that can be used separately or in combination for interaction and control of a VR environment. In this study we use the standard off-the-shelf Xbox [96] controller shown in Fig. 3.1 (a) that comes prepacked with the Oculus Rift[105] bundle (see Fig. 3.1 (b) and Fig. 3.1 (c)). This controller-style has achieved very high adoption in gaming industry; its design is ergonomic and easy to learn. As one example of wider adoption, the US Navy recently adopted the use of a Xbox controller to operate the periscope on nuclear-powered submarine[1].

Here, this Chapter present a VR aerospace design environment with a focus on dimension reduction. I García-Hernández et al. [42] suggests that three elements are especially promising for a VR-based approach: 1) *integration of multiple 2D graphs for 3D data* [42]; 2) *3D parallel coordinates* [42] (see Tadeja et al. [150, 151]); and 3) *visualisation of complex graphs* [42]. In this study, we loosely follow (1), but with a key difference: we use subspace-based dimension reduction to generate the 3D graphs for high-dimensional data.

Applications in Design

Our dimension reduction results and case study is based on the work undertaken in Seshadri et al. [129]. Here the authors study the 25D design space of a fan blade using the quadratic active subspaces recipe detailed in Section 3.1.1. Towards this end, we used the design of an experiment with $N = 548$ 3D RANS computations with different designs; the design space used in this study included five degrees of freedom specified at five spanwise locations. These degrees of freedom comprised of an axial displacement, a tangential displacement, a rotation about the blade's centroidal axis, leading edge recambering and trailing edge recambering, specified at $0, 25, 50, 75$ and 100% span. Thus, we obtained values of the efficiency and pressure ratio for each design vector $\mathbf{x}_i$. These are two important output quantities of interest

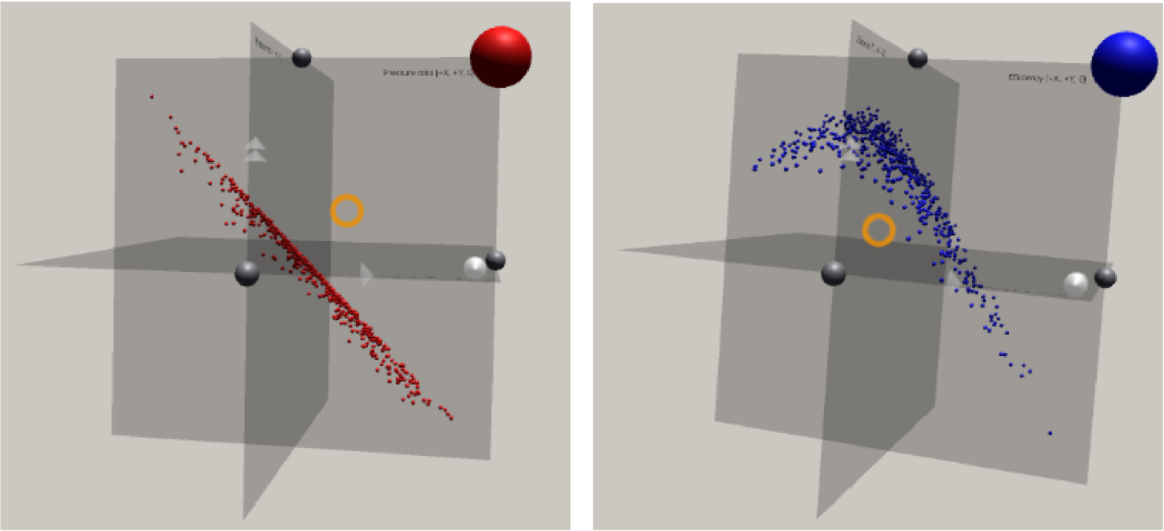
[1] https://www.digitaltrends.com/cool-tech/navy-xbox-controllers-attack-submarines



(a) Pressure ratios

(b) Efficiency

Fig. 3.2 Sufficient summary plots of (a) Pressure ratios; (b) Efficiency, for a range of different computational designs for turbomachinery blade, obtained from a design of experiment study. Based on work in Seshadri et al. [129].



(a) Pressure ratios

(b) Efficiency

Fig. 3.3 Sufficient summary plots of (a) Pressure ratios; (b) Efficiency presented in Fig. 3.2 as seen by the user in the AeroVR environment. Based on work in Seshadri et al. [129].

in the design of a blade. By studying the eigenvalues and eigenvectors of the covariance matrix for these two objectives the authors were able to discover a 1D ridge approximation for the pressure ratio of a fan and a 2D ridge approximation for the efficiency. These sufficient summary plots are shown in Fig. 3.2. There are a few important remarks to make regarding these plots.

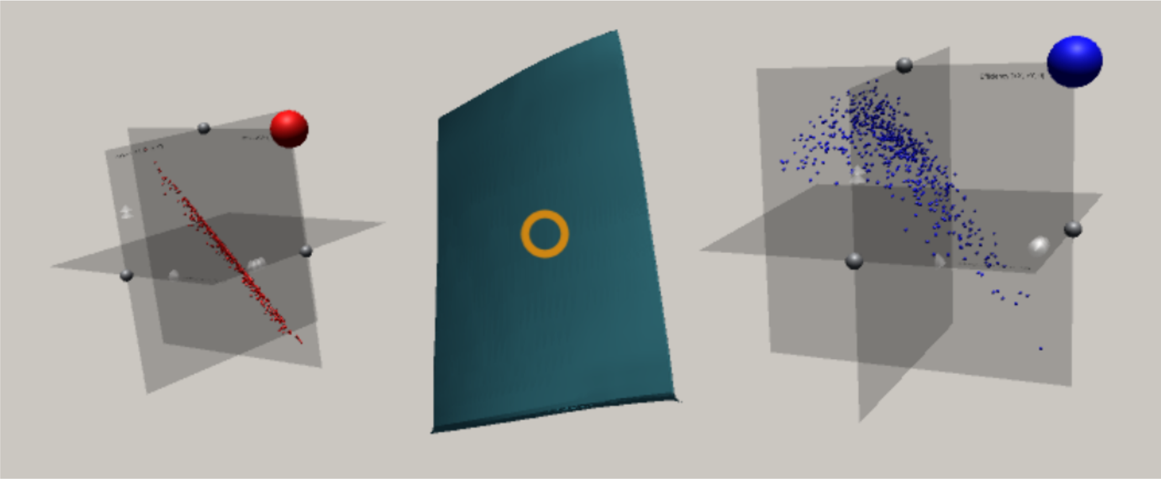


Fig. 3.4 The user's field-of-view: the right-hand plot can, for example, show the lift coefficients whereas the left-hand side plot can contain the drag coefficient values. The nominal blade geometry is visualised in the middle between the two plots. The orange circle is a cross-hair signalising where user is looking at the moment.

For the pressure ratio sufficient summary plot, shown in Fig. 3.2 (a), the horizontal axis is the first eigenvector of the covariance matrix associated with the pressure ratio, $\mathbf{u}_1$. For the efficiency sufficient summary plot, shown in Fig. 3.2 (b), the two horizontal axes are the first two eigenvectors of the covariance matrix associated with the efficiency $[\mathbf{u}_1, \mathbf{u}_2]$. It is important to note that the subspaces associated with efficiency and pressure ratio are distinct.

The sufficient summary plots above permit us to identify and visualise low-dimensional structure in the high-dimensional data. More specifically, these plots can be used in the design process as they permit engineers to make the following inquiries:

- What linear combination of design variables is the most important for increasing / decreasing the pressure ratio?
- How do we increase the efficiency?
- What are the characteristics of designs that satisfy a certain pressure ratio?
- What are the characteristics of designs that satisfy the same efficiency?

We use these sufficient summary plots in a bespoke VR environment (see Fig. 3.3). Our high-level objective is to ascertain whether it is possible to leverage tools in VR in conjunction with parameter-space dimension reduction to facilitate better design decision-making and inference. To achieve this goal, we seamlessly integrate the aforementioned sufficient summary plots with the 3D geometric design of the blade, i.e.,

$$\text{pressure ratio} \rightleftharpoons \text{geometry visualisation} \rightleftharpoons \text{efficiency}.$$

In other words, as the user selects a different design—by selecting a suitable level of performance from the sufficient summary plots—they visualise the geometry of the blade that yields that performance. Moreover, they should be able to compare this geometry with that of the nominal design. We clarify and make precise these notions in the forthcoming subsections.

3.1.3 Function Structures

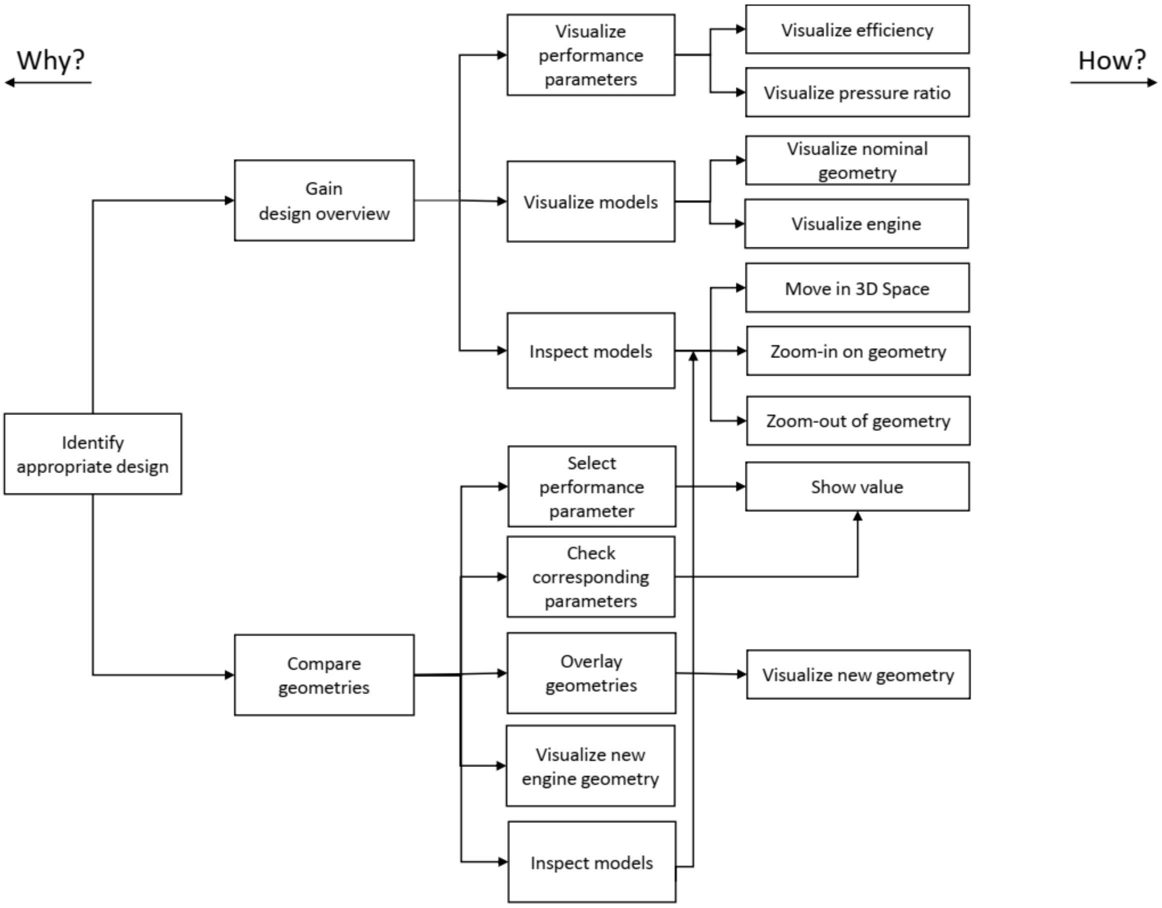


Fig. 3.5 A functional model of the interactive system.

We model the function structures of the system using *Function Analysis Systems Technique* [134] (FAST). Fig. 3.5 shows the function structures of the VR visualisation environment for an aerospace design workflow with dimension reduction. The FAST-diagram in

Fig. 3.5 models the level of abstraction on the horizontal axis and function sequence on the vertical axis.

3.1.4 System Structure

We model the system internal structure by observing the internal flow of signals between the individual system elements. The visualisation consists of four distinguishable parts: (A) the user who is responsible for all the actions of the system once the data had been loaded and visualised, (B) efficiency and pressure-ratio 3D scatter plots, (C) blade model, and (D) engine geometry model. The signals are usually bi-directional and can introduce a *chaining effect*. For instance, when user is gazing over an interactive object, which is internally facilitated by the ray-tracing, the object highlights itself, that is, the user receives a return feedback signal in the form of a visual clue. Moreover, selection of a data point on the scatter plot through an implementation of the *linking & brushing* interaction technique, leads to a selection of the mapped data point on the other scatter plot and visualisation of a new geometry overlapping with the nominal shape. The signal flow analysis is presented on Fig. 3.6. The main signal flows are decomposed into:

(A) **The user:** The user interacts with the system using a combination of gaze-tracking and ray-tracing. This works as follows. Gaze-tracking is achieved with the help of a cross-hair in the middle of user's field of view, placed a certain, fixed distance along the camera's forward direction. Rays extending from the cross-hair are constantly checked for intersection with other interactive objects i.e. data points on the scatter plots. If such an interaction occurs, the object automatically highlights, providing a signal to the user that it can be interacted with. The way in which the user directly receives signals from other parts of the visualisation is unidirectional, that is, a user's action results in a visual response. The way the user interacts with other objects is through a combination of gaze-tracking and ray-tracing (i.e., an orange cross-hair, see Fig. 3.4 and Fig. 3.8) as well as actions invoked with a tap of a button (see Fig. 3.1).

(B) **3D scatter plots:** The scatter plots receives signals from the user through a mixture of gaze-tracking and ray-tracing inputs combined with the tap of a button on the controller. This is reflected back to the user by, for example, highlighting scatter plots elements, such as data points or movement selectors, that are being gazed over or changed their colour after selection. In turn, this action invokes unidirectional changes in the visualised blade geometry and the turbofan engine.

(C) **Blade geometry model:** The blade geometry visualisation receives signals from both scatter plots by the user performing a selection of a data point on any of the plots, which automatically visualises the new blade geometry. Moreover, even though the user cannot directly influence the geometry, by using the movement and manoeuvring techniques in the system, the user can inspect the geometry by zooming in on its internal and external surfaces. Hence the relation between the scatter plots and the blade is unidirectional, whereas the relation between the user and the blade model is bidirectional. Furthermore, once the new blade has been visualised, the visualisation of the hub with blades in the engine model simultaneously changes as well. This can be thought of as another unidirectional relation as it cannot happen the other way around.

(D) **Engine geometry model:** Once the new geometry shape is selected by the user, the blade row with the series of blades embedded in the engine model is automatically replaced. This change is immediately visible to the user providing visual feedback.

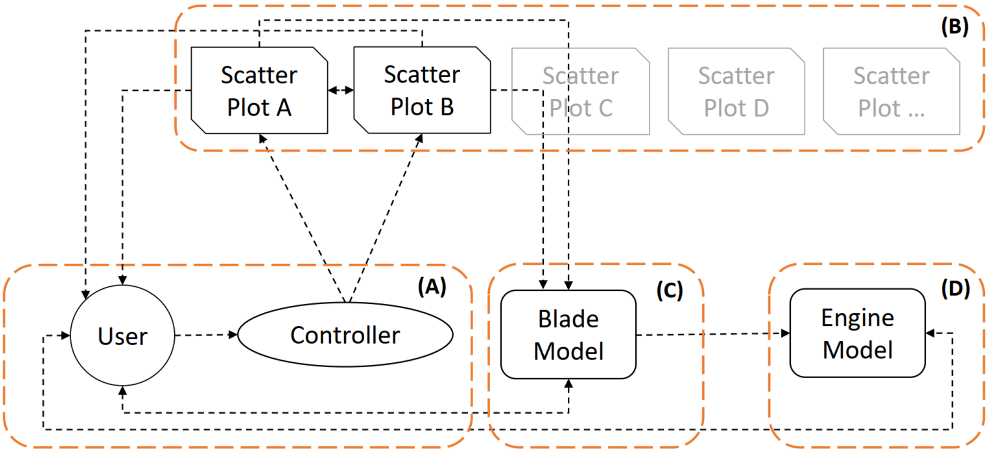


Fig. 3.6 The diagram shows how the signals are flowing within the system between its four main components: (A) the user grouped together with a controller used for user input; (B) a set of performance parameters visualised as the 3D scatter plots, in this case, efficiency and pressure-ratio 3D scatter plots; (C) blade geometry model; and (D) complete engine geometry model.

3.1.5 Task Analysis

From the limitations imposed by the current state and understanding of the VR environment and from our own analysis of the system achieved by the FAST analysis (see Fig. 3.5) we identified two primary, high-level tasks:

T1—Gaining design overview: The system should permit the user to easily gain an overview of the entire design space i.e. possible blades geometries together with their associated performance parameters.

T2—Compare geometries: The system should permit the user to easily compare the nominal blade geometry with the one associated with a particular set of performance.

These two main tasks (i.e. *T1* and *T2*) were supported and augmented by a number of low-level tasks:

T3—Movement and interaction: Due to the nature of spatial, 3D immersive workspace provided by the VR environment, this task has a dominant and a supporting role with respect to all the other tasks. Movement, manoeuvring and interaction are achieved through the gaze-tracking and with the help of a gamepad controller. All movement facilitated by either the joystick [*J*] or triggers [*T*] (see Fig. 3.7) takes place with respect to the users gaze (see orange cross-hair on Fig. 3.4). Moreover, the user can interact with an object through gazing over an object and tap of a button (see Fig. 3.7). Zooming in or out on a part of the visualisation is also achievable by the user's movement in the virtual space.

T4—Visualisation of performance parameters: The performance parameters, such as efficiency and pressure ratio that were used in our case, are visualised as an interactive 3D scatter plots floating in the 3D space. These can be freely moved, rotated about each of the main axes and implements the *linking & brushing* interaction technique i.e. changes in one scatter plot are simultaneously reflected on the other scatter plot and blade and engine visualisations as well. This task mainly supports *T1*.

T5—Visualisation of blades and engine models: The nominal blade geometry and associated engine visualisation are immediately visible at the start of the visualisation. Once the new geometry is selected, the nominal blade renders semi-transparent and the shape of the new blade is superimposed over it. Moreover, the hub with a row of blades are substituted with the new geometries in the engine model.

T6—Models inspection: The inspection of the changes in the engine visualisation and the blade geometry itself can be made through the *T3* task.

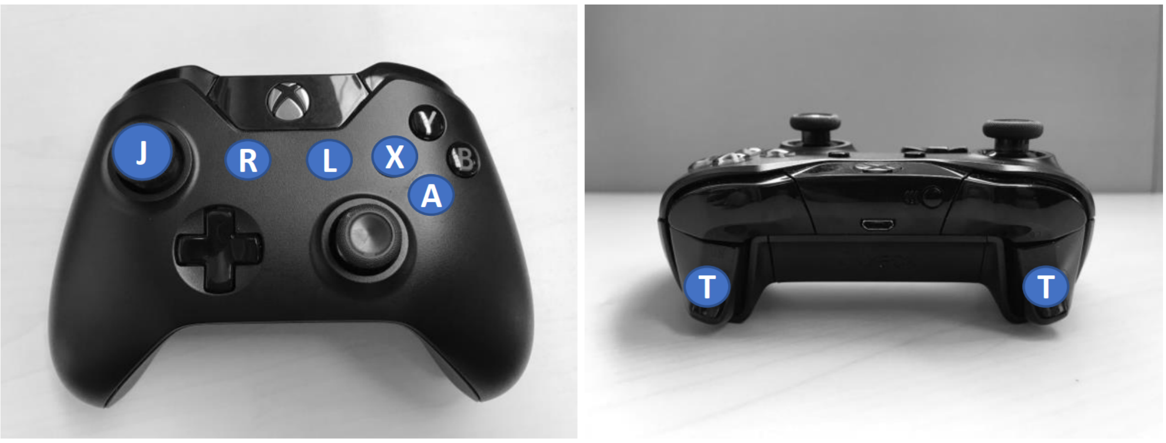
3.1.6 Visualisation Framework

The visualisation framework is built using Unity3D—one of the most widely used game engines with built-in VR development support. Both of the two mainstream VR headsets provide supporting packages developed natively for Unity3D, which substantially speeds up the development process. This software is built on top of the *Unity VR Samples pack* [166] and uses the *Oculus Utilities for Unity* [106] package as well as parts of the Unity asset [47].

In addition, we use the asset store available for the Unity3D game engine, which contains many VR-ready tools and supporting packages.

A survey by Wagner et al. [170] highlights that game engines "*do not support any data exploration*"[170] techniques. In other words, these features have to be designed and implemented from scratch. To allow user interaction with data we use an Xbox Controller connected with the laptop via USB cable (see Fig. 3.1 (a)) in combination with gaze-tracking through a cross-hair which moves with the user's head and is placed straight from the camera (visualised as an orange cross-hair, see Fig. 3.4, Fig. 3.8 and Fig. 3.11).

3.1.7 Interaction and Movement



(a)

(b)

Fig. 3.7 The Xbox controller: (a) shows the top view with the left-hand joystick [J] used to control the 2D movement on the X-Z plane whereas (b) shows the front view with the two triggers [T] responsible for vertical movement along the Y axis. The other action buttons indicated in (a) have the following meanings: [R] for reload of the visualisation; [L] for loading next dataset; [A] for selection of an interactive item; and [X] for moving or rotating the scatter plots.

The interaction is designed around gaze-tracking in combination with the standard buttons on the Xbox[96] controller (see Fig. 3.7). Supported interactions are mapped as follows:

- Left-hand joystick and [T] buttons (see Fig. 3.7): Triggers movement along the X-Z plane and movement along the vertical axis respectively, right-hand [T] is assigned to "up" and left-hand [T] is "down". The movement in the X-Z plane is always with respect to the user's gaze. This manoeuvring combination permits the user to move in any direction and in any position in 3D space. The user moves with constant velocity and with fluid movement to ensure the user is receiving continuous closed-loop visual feedback on their changing position in relation to the surroundings.

- Action button [A]: Selects an interactive element, such as a scatter plot rotation and movement selector, or a data point (see Fig. 3.11). Objects highlight themselves when the user's gaze, as indicated by a cross-hair, is on them. Double-tapping on the [A] button selects the highlighted object.
- Button [X]: When tapped after the selection of a scatter plot point, it will re-position the point to a certain distance towards the user's present gaze direction. Furthermore, if the rotation selector is active, selecting this button will initiate the scatter plot's rotation over 90° based on the current direction of the user's gaze.
- Button [R]: Resets the visualisation and all its associated elements to their original state.
- Button [L]: This button loads the next dataset: a new set of performance parameters and associated blade geometries.

3.1.8 Blade and Engine Visualisations

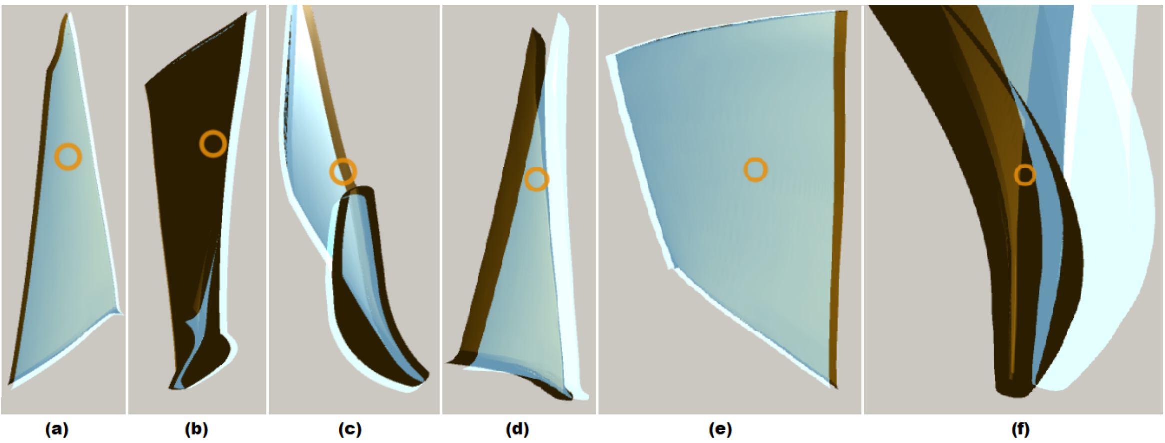


Fig. 3.8 By selecting any data point on any plot, the user can immediately observe the blade's geometry associated with this particular design. Moreover, the user can observe and compare the differences between the nominal and perturbed geometries as the former is kept rendered as a semi-transparent shape overlaying the latter. As users can freely manoeuvre in 3D space they can visually inspect the entire blade from any direction and zoom in on any of its parts, as shown in (a–f).

As alluded to previously, the central artefact in our VR environment is the geometry of the designs. Our virtual environment contains as many geometries as there are data points, resulting in a total of 548 stereo lithography (STL) files. Hence, whenever a data point is selected on one of the plots, the accompanying shape is instantly visualised. To provide the user with a quick and an effective way of comparing the new perturbed design, the nominal geometry is still kept visible and rendered as a translucent object, as can be seen in Fig. 3.8. This solution, combined with unlimited movement dexterity, allows the user to visually inspect and observe any differences between the two geometries. Furthermore, by simply changing their position, or by tilting their head (thereby changing the rotational angle), the user can zoom-in and zoom-out on any of the blades parts for a close inspection as presented in Fig. 3.8 (f).

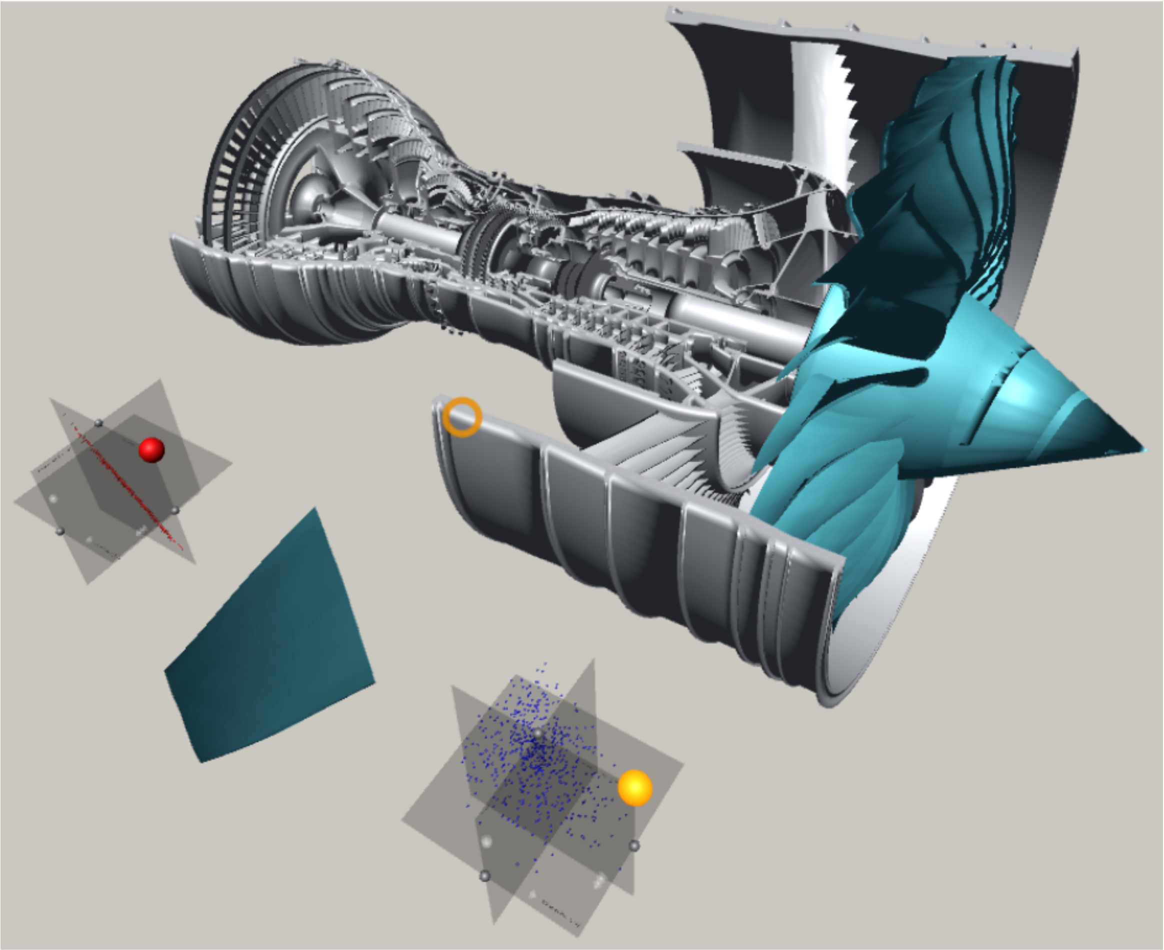


Fig. 3.9 The entire visualisation as it is seen by the user with the complete engine model in the back and the two 3D scatter plots and the nominal blade geometry (in blue) in front. The hub with a series of blades is also shown (in blue).

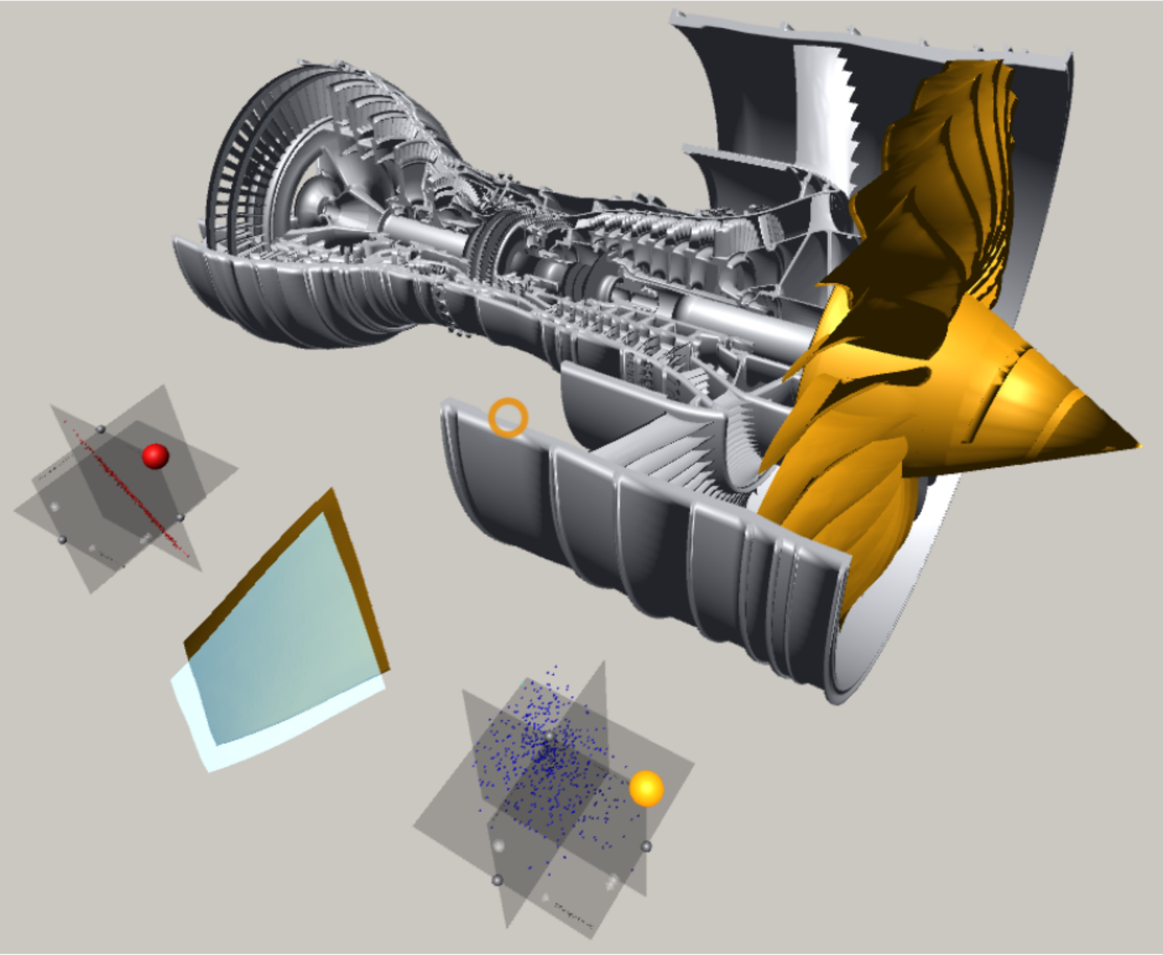


Fig. 3.10 The same view as in Fig. 3.9 with a single data point selected on one of the scatter plots (the one on the bottom-right). The nominal blade geometry was rendered as semi-transparent shape with a new geometry superimposed on top of it. The engine hub with a new series of blades is also shown (in tan).

The visualised engine model [133] (see Fig. 3.9) consists of six independent parts including the hub with connected blades. When a new blade geometry is being investigated by the user, the blades visible in the engine are automatically replaced as well. Due to limitations imposed by the used CAD model itself, it was not possible to substitute the blades individually, thus the entire hub with the series of attached blades could be replaced altogether which is signalled to the user by changing colour of this entire part (see Fig. 3.10).

3.1.9 Sufficient Summary Plots

The key element in the framework is the sufficient summary plots; they are visualised as three fixed-size, axis-aligned translucent orthogonal rectangles. The data points are scaled so the values of their respective coordinates are within the range of the translucent surfaces. When any of the spheres denoting a data point is selected, the marker lights up and switches to a selection colour (light green). Moreover, as we have a 1:1 mapping between the plots, the corresponding design on the other plot is selected. Furthermore, a number of semitransparent cones[2] was embedded into these sufficient summary plots to denote their axes: one for the X-axis, two for the Y-axis and three for the Z-axis. Selection of a shape with its pointing tip has an additional advantage—the orientation informs the user of the positive side of a given axis. Each selection can be reverted by double-tapping the [A] button while gazing over it.

The 3D spheres in the plots were used to denote both the data points and various selectors' markers. Using shape perception has a long-standing application history for VR-based visualisations. For instance, Ribarsky et al. [114] used simple 3D shapes such as cones, spheres and cuboids in their system. They also highlight that glyphs with their intrinsic characteristics, such as *"position, shape, colour, orientation and so forth"* [114] are very useful when visualising complex datasets.

Initial Placement and Repositioning

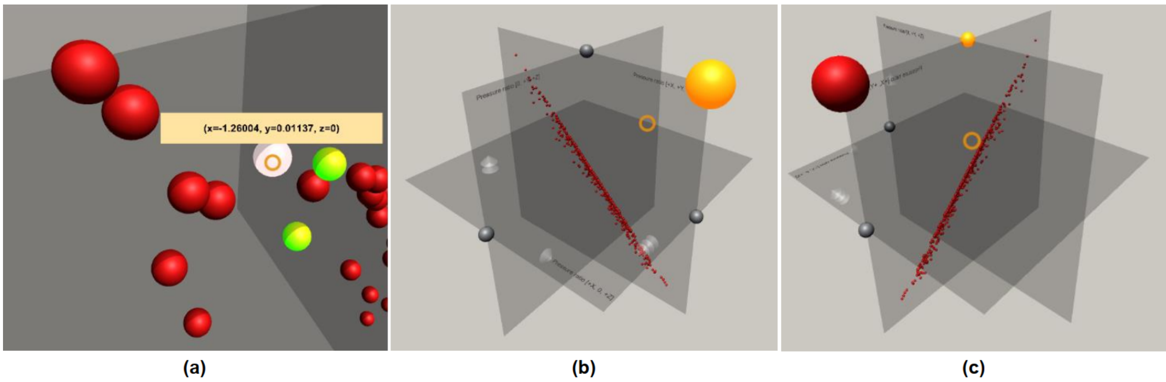

(x=-1.26004, y=0.01137, z=0)
Pressure ratio [0, +Y, +Z]
Pressure ratio [+X, +Y
Pressure ratio [+X, 0, +Z]
(a)
(b)
(c)

Fig. 3.11 The scatter plot of the Pressure ratios data (see Fig. 3.2(a) and Fig. 3.3(a)) as it is seen by the user. Figure (a) shows a user's gaze (orange cross-hair) hovering over a data point which instantly displays the associated values (e.g. its coordinates). Visible, formerly selected points (in green) are reflected on the other scatter plot and the associated geometries are also shown. Figure (b) shows the same plot from a distance. The highlighted spheres (in orange) are the movement selectors: if the user gazes at any point in space and taps the [X] button on the controller the plots will be translated towards that point in space. Figure (c) shows a rotation by 90° towards the user along the Y-axis with the axis rotation selector highlighted (in orange). Only a single rotation selector can be active at once across all the scatter plots.

[2] W. Kresse, used under CC BY-SA 3.0; http://wiki.unity3d.com/

The two scatter plots—one for pressure ratio and another for efficiency—are automatically positioned on both sides of the blade, which in turn is positioned in front of the initial user's field of view; see Fig. 3.4. The plots are placed at the same, pre-configured distance from the user, at a roughly 45° angle from the X-axis.

To ensure that the user does not feel constrained in a nearby region and to make better use of virtually infinite 3D space provided by the VR-environment, users are provided with the possibility of moving the scatter plots. The interaction occurs via gaze-tracking and the *select & move* metaphor. Every scatter plot has a colour-coded interactive sphere, that is, a selector (see Fig. 3.11 (b)), attached to it in the right-hand top corner. When the user's gaze hovers over it, the selector automatically highlights it and, if selected by double-tapping the [B] button, changes its colour to orange (see Fig. 3.11 (b)). If the user presses the [X] button while a selection is active, the scatter plot is re-positioned at a certain distance towards the point determined by the user's current gaze. If the button is held the plot will follow the cross-hair's movement.

It is also possible to move both plots at once if more selectors are simultaneously active. In such a scenario, to keep the current relative position of these scatter plots, a barycentre $B = (x_B, y_B, z_B)$ of all these objects is calculated using the formula:

$$B = \frac{1}{N}(x_B = \sum_{i=0}^{N} x_i, y_B = \sum_{i=0}^{N} y_i, z_B = \sum_{i=0}^{N} z_i). \tag{3.8}$$

where (x_i, y_i, z_i) are coordinates of the plots' individual centres. This point is moved along the forward vector from the camera in the same manner as in the case of a single scatter plot. Selected objects are then grouped together and displaced with respect to the new position of the barycentre whilst simultaneously keeping their internal (current position with respect to the local axes) and external (axis-alignment of surfaces) rotations.

Scatter Plot Rotation

The scatter plot can be rotated in 90° steps about one of the three main axes. This is achieved by double-tapping the [A] button while gazing over one of the three rotation selectors (see Fig. 3.11 (b-c)). A further press of the [X] button will result in a rotation of all the plot's components, including data points, rotation and movement selectors and axis cone markers. If users are situated in such a way that their gaze is located exactly in front of the active axis selector, the rotation will occur towards the direction provided by the camera's forward vector. Similar to the movement, the plot's orientation in the global coordinate system will not be affected.

Relationships Between the Visualisation Elements

The data points on one of the plots are correlated in a 1:1:1 (one-to-one-to-one) manner onto the other plot and vice-verse. Moreover, each of the data points is mapped onto one-and-only-one unique blade design. Therefore, whenever a marker is selected on one of the plots, the system will automatically highlight the corresponding data point on the other plot and switch the visualised blade onto the new, corresponding shape. Furthermore, the nominal shape will be kept as a translucent point of reference (see Fig. 3.8) that overlays the new design to show the user how, where, and to what degree, the new shape differs from the nominal one.

Labelling

Both the data points and the axes are automatically labelled. In case of the latter, the strings embedded into the edges of the semitransparent rectangles denoting the axes are read directly from the input text file (see Fig. 3.11 (a)). The labelling of the data points is only visible once the user is hovering with his or her gaze over the marker (for example, a sphere) and disappears once the user looks at another point, or other parts of the visualisation (see Fig. 3.11 (a)). Furthermore, the small box with the values associated with the point (in this case its coordinates) is always rotated towards the user and follows their gaze. It is rendered on top of any other visualisation elements as seen in Fig. 3.11 (a).

3.1.10 Interface Verification

The VR aerospace design environment described in this study is still at an early stage and the objective of this Section is not to present a complete solution but to demonstrate potential benefits of VR for aerospace design.

Here, we verify the fundamental usability of the system using two formative evaluation methods. First, we assess the usability of the system using Nielsen's [102, 101] guidelines. Second, we use the cognitive dimensions of notations framework [45, 46] to reason about the expressiveness of the system.

Usability

- *Visibility of system status*: The system provides immediate feedback to users in response to their actions. For instance, whenever a user's gaze hovers over an interactive object (such as, for example, movement and rotation selectors or a data point) it is instantly highlighted. In addition, once an object is selected, the object also changes its colour in response. In addition, following the selection of a data point, its corresponding data points on the other plot are also simultaneously selected and the accompanying geometry is loaded automatically. This ensures the user remains synchronised with the system's current state. Finally, whenever the user's gaze is hovering over an object, a gaze-locked text is also displayed to the user, which reveals values associated with this particular data point.

- *Match between the system and the real-world*: First, the system uses the cross-hair concept which is well known in the real-world to focus and help guiding the users' gaze on objects placed directly behind or near it. Moreover, the 3D scatter plots were designed to immediately resemble their two- or three-dimensional desktop-based counterparts. In addition, initially a user observes all the main elements of the visualisation in the field-of-view placed at roughly the same height and direction as how they would be perceived in the real-world if they were visualised on standard computer displays.

- *User control and freedom*: The user can either load the new set of data or reload the entire visualisation with a click of a button. No direct "undo" and "redo" actions were directly implemented, however, users can always deselect any object or redo the last executed operation. Moreover, using a combination of the gaze-tracking and controller-based interaction, users can locate themselves at any position in 3D space.

- *Consistency and standards* and *Flexibility and efficiency of use*: As the VR in its current, almost fully immersive form, is a fairly recent development, the technology itself, not to mention its main applicability areas or interaction design principles, is not yet fully understood. However, the system design is as consistent as possible, for instance, all interactive objects can be (de)selected using exactly the same method.

- *Error prevention*: Measures to prevent the user from errors, such as missing or broken input data (for example, geometries and data points), are directly built into the system. As there is a 1:1:1 mapping between the system elements (data points) on the two plots and the geometries, missing any of the elements would lead to omitting this particular entry from the visualisation and detailed information of such event being written into the log file. Hence, the main error-prone conditions are eliminated. In addition, the system incorporates certain constraints, such as a user cannot have more than a single axis-rotation selected at the time—the new selection will automatically deselect any previously activated selector. This prevents an error caused by the system being unable to recognise about which axis the scatter plot should be rotated.

- *Help users recognise, diagnose, and recover from errors*: There are not many errors that user can commit, assuming the dataset is correct. The countermeasures against plotting incomplete data are built-in into the system. However, due to the nature of the visualisation, erroneous blade geometries or any anomalies in their shapes will be detected by the user in a close-up inspection possible through the mixture of movement and manoeuvring in the 3D space. The same can be said about the 3D scatter plots where user is able to rotate them and see them from every direction and can zoom in and zoom out from any data point using the same techniques.

- *Recognition rather than recall* and *Aesthetic and minimalist design*: The blade's geometry is visualised as a replication of its physical appearance in the real-world. Moreover, both plots use volumetric glyphs to denote the points which reassembles the 2D scatter plots versions (see Fig. 3.11). Previously selected data points are also highlighted. Furthermore, the system was designed to be minimalistic—only the efficiency and pressure ratio plots together with the geometry visualisation are included to avoid overloading the user with information. Hence, for instance, the values of the data points are not initially visible, however, the scatter plots offer a possibility of gaining a high-level understanding of the data at a first glance. More detail of each individual point is available on an on-demand basis when the gaze cursor hovers over a data marker.

- *Help and documentation*: A succinct single page documentation sheet describing the interaction techniques is provided to the users.

Expressiveness

The expressiveness of the system is here analysed using the cognitive dimensions of notation framework [45]. This framework provides a vocabulary for analysing the possibilities and limitations of an interactive notational system. Below we articulate how the keywords in this vocabulary maps onto the expressiveness of the system.

- *Closeness of mapping*: The visualised geometry is a detail mapping of how would the blade would look like in the real-world.

- *Consistency*: All interactions are consistently designed and all interactive objects have consistent interaction qualities, such selectors.

- *Diffuseness / terseness*: The number of used symbols is minimised by only using sphere-like markers with their characteristics (such as colour, size and relative placement) to denote all the interactive elements of the visualisation.

- *Error-proneness*: Errors are prevented using prevention mechanisms against error states, such as selection of multiple rotation-selectors.

- *Hard mental operations*: Cognitive load and mental demand is kept to a minimum with straight-forward interaction methods and use of comparative visualisations.

- *Hidden dependencies*: As there is a 1:1:1 mapping between the visualisation elements all the interdependencies are easily observed by the user since selection of a data point on one of the plots leads to a simultaneous selection of a corresponding data point on the other plot as well and the visualisation of the associated geometry.

- *Juxtaposability* and *Visibility*: All three main parts of the visualisation, that is, the efficiency and pressure ratio plots together with the blade's geometry, are initially placed next to each other. Furthermore, the plots can be freely rearranged in the space as a group or individually. Moreover, the selection of any data point in a plot is automatically mapped on to the other plot as well and the corresponding geometry is immediately visible. In addition, selected data points are clearly visible through change in colour and luminosity.

- *Premature commitment*: The user's workflow with the system is flexible and a user is free to initially inspect the nominal geometry, any or both plots, or to immediately select a data point.

- *Progressive evaluation*: Since all the user's actions result in immediate visual feedback (closed-loop interaction) the progress of the visual analytics task can be evaluated by the user at any time.

- *Role-expressiveness*: The individual roles of the three components, that is, the efficiency and pressure-ratio plots and the blade's geometry, are clear from the beginning, especially if the system is used by a domain expert.

3.1.11 Discussion

This Section introduced the *AeroVR* system for aerospace and aeronautical design facilitated by the fully immersive, VR-based environment. In order to developed our system, I have carried out the FAST [134] and task analyses to understand what functionality is needed in order to fulfil its primary goal of aiding the aerospace design processes. I have also verified the interface from two perspectives: usability [102, 101] and expressiveness [45] using two well-established heuristics techniques.

I resorted to these methods instead of conducting user studies because the application of VR in aerospace design is still underexplored. This means that the design space is very large. Thus, I needed to first trim it in order to understand the limitations of the VR equipment before it was sensible to organise troublesome, and time-consuming user experiments. Furthermore, the results gathered in such studies could be questionable and of poor quality without any prior knowledge of the particular use case.

Nonetheless, thanks to the heuristic evaluation, I have reason about and developed the fully functioning system for aerospace design. Furthermore, we have identified two key, high-level tasks for such systems. These were: *T1: gaining the design overview*, and *T2 comparing the nominal geometry* to the ones associated with a specific performance parameters. The former task *T1* was achieved through a mixture or visualisation (e.g. blade and engine geometry, and scatter plots) and lower-level tasks (e.g. movement and interaction). Whereas, the latter task *T2* was achieved through visualisation of the overlaying semi-transparent blade models superimposed over the nominal blade's geometry (see Fig. 3.8).

3.2 Digital Twin Assessments in Virtual Reality

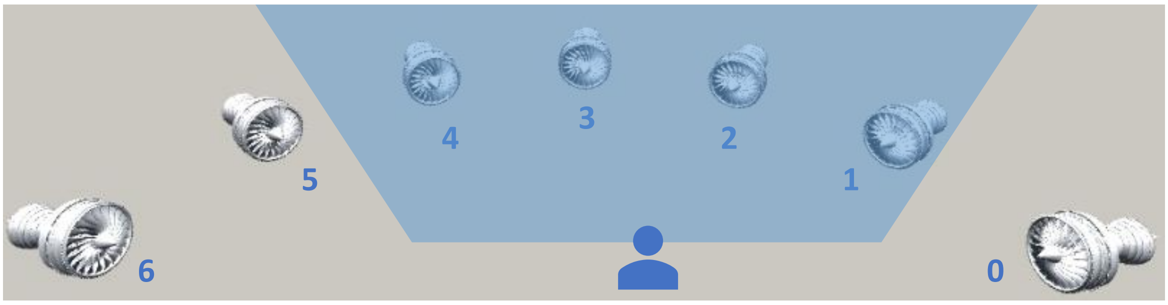


Fig. 3.12 The array of aeroengines CAD models arranged in a counterclockwise ordered around an arc in equal steps centred around a user's avatar. Assuming the engines are not moved or the user's avatar does not change its position, at most four engines will be visible in the user's field of view.

Digital twins are multi-scale, multi-physics, probabilistic computational representations of real engineering assets [43]. They are designed to characterise and model key input-output dependencies, based on data acquired from sensors on the monitored assets and physics-based knowledge that is encoded in varying fidelities of computational tools. These assets can range from bridges and buildings to jet-engines and airplanes. More physically representative digital twins empower engineers to gain better insight into their system's operation, offering the ability to make more well-informed decisions. This newly emerging field of *data-centric engineering* (see [14, 172, 130, 128] and references therein) is a unique pairing of advanced computational statistics with engineering principles and is ushering in a new era of digitisation across all branches of engineering. The focus in this Section will be on such assets within the aerospace sector specifically tailored for aeroengines.

The increase in digitisation within the aeroengine sector is concurrently taking place with an increase in new human-computer interaction techniques. This should not come as a surprise, as improvements in the former can be catalysed by the latter. One propitious human-computer interaction pathway has been that of virtual reality (VR), where bespoke 3D environments can be constructed, and users can leverage a range of different interaction paradigms, including visual, audio and even haptic feedback. It is thus natural to think of VR as a well-suited environment for (1) interactive visualisation; and (2) analytical reasoning for complex multidimensional data—in this work specifically for aeroengine operation.

Bringing together these two avenues for cognition, we can think of having a *Visual Analytics* system as first defined in [161]. These ingredients, coupled with the newest advances in the field of so-called immersive technologies, which includes not only VR, but also Mixed Reality (MR) and Augmented Reality (AR), have given rise to the new research field of *Immersive Visual Analytics* [19]. This research field uses the immersive environment as the main medium for interactive data visualisation and consequently investigates how to effectively design (and use) such systems. However, there are still a number of obstacles that need to be overcome for mass adaptation of such tools. For starters, there is the need of demanding computing resources required to support costly VR infrastructure [105]. Then there are occlusion effects [137], which plague nearly all 3D interactive paradigms. Finally, there is the challenge of demonstrating that VR promotes more efficient visualisations that lead to more effective reasoning, when compared with today's existing modes of user-data engagement. That said, some strides have been made in abating these barriers to entry and acceptance. For instance, with regards to infrastructure, recently, a new wave of wireless, standalone VR headsets have been introduced (e.g. Oculus Go or Oculus Quest [105]) which diminish the barrier to entry in terms of the costs for the novice users. Additionally, new interaction methods used to interact with VR-based environments are currently being investigated, including coupling gaze and hand tracking [141].

These developments set the stage for our work. We study the health of a fleet of aeroengines (see Fig. 3.12) in a bespoke VR environment that seamlessly blends computer-aided design (CAD) models of the engines with a series of interactive analytical graphs that reveal the performance data of each engine. Our environment, tailored for the aeroengine engineer, is designed to permit (i) rapid comparisons between engines; (ii) facilitate fault detection; and (iii) enable a greater understanding of the location of various sub-systems, components and the interactions between.

3.2.1 Aeorengine Visualisation System

In what follows we detail particulars of our aeroengine visualisation system. The whole visualisation was written using the Unity game engine [165].

Apparatus

The experimental and development setup consisted of NVIDIA GeForce GTX 1080 GPU and an Intel i7-7700K 4.20 GHz CPU working under the Windows 10 Pro 64bit operating system with 32 GB of RAM. The VR environment used an Oculus Rift [105] head-mounted display (HMD). Hand-tracking was permitted with the Leap Motion sensor [84] attached in front of the HMD, and gesture recognition was developed using the Leap Motion SDK [84].

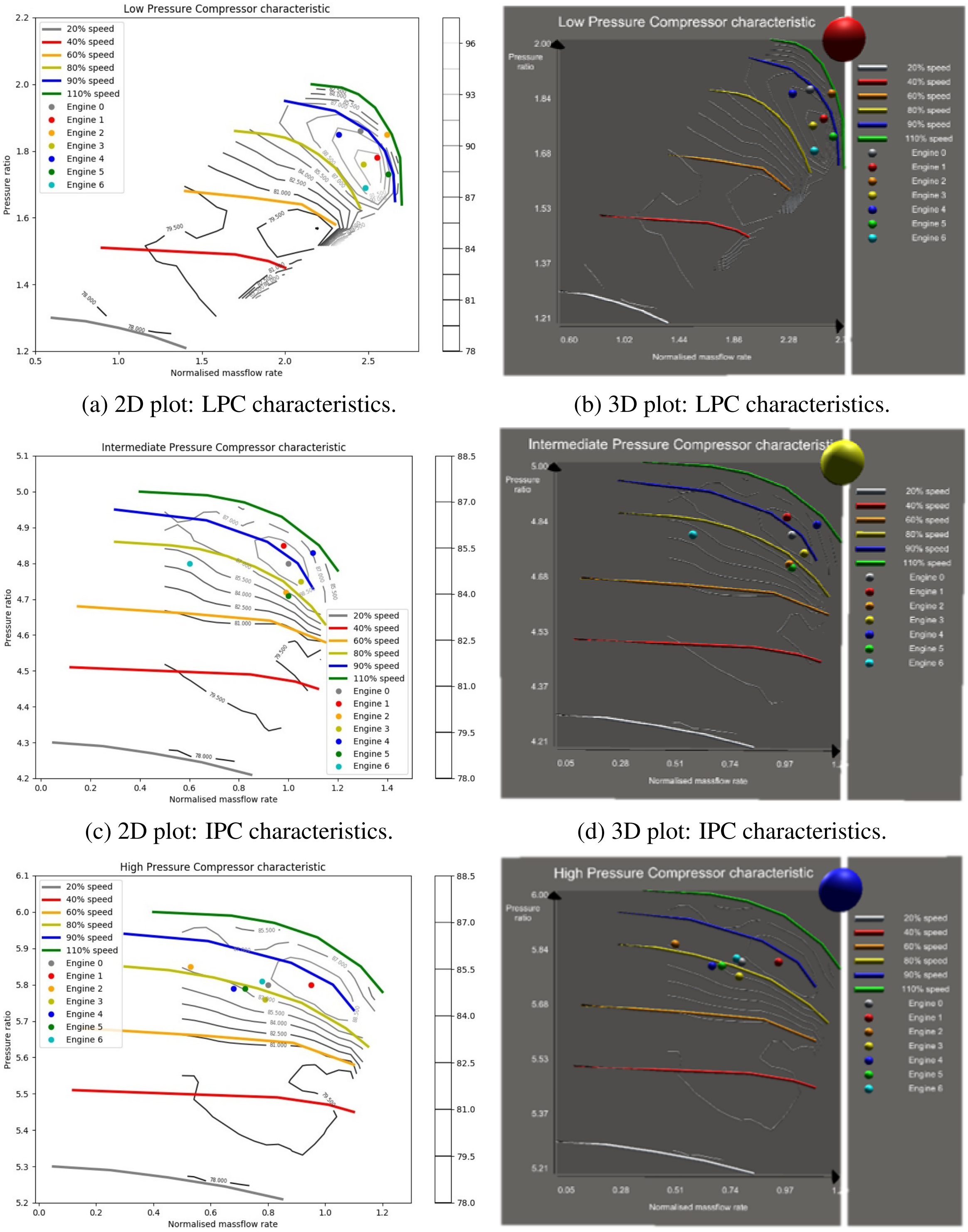
Aeroengines Models

The system consists of an array of an interactive aeroengine CAD models [57] arranged in equal steps along a semi-circular arc in front of the user's initial position and gaze direction, as shown in Fig. 3.12. In our case, for testing, we use seven aeroengine models. However, the number of aeroengine models can be easily adjusted depending on a particular user's needs and wants. Each of the models consist of eleven subsystems, including low (LPC), intermediate (IPC) and high pressure compressors (HPC). These three compressors—standard subsystems for a three-shaft aeroengine—form our main subsystems of interest.

The full list of the available engine sub-components that do not contain any sub-parts themselves consist of these eleven elements: (i) casing; (ii) low pressure compressor (LPC); (iii) high pressure compressor (HPC); (iv) intermediate pressure compressor (IPC); (v) low pressure turbine, shaft and nozzle; (vi) fan; (vii) nose cone; (viii) high pressure shaft; (ix) intermediate pressure shaft; (x) intermediate pressure turbine; (xi) combustor and high pressure turbine.

Compressors Characteristics Plots

Each of these three compressors are accompanied with interactive performance maps that characterise their nominal working condition. These maps plot the pressure ratio and efficiency as a function of shaft speed and inlet non-dimensional massflow rate for the numerous engine sub-systems, as can be seen in Fig. 3.13. The first column of the plots



(a) 2D plot: LPC characteristics.

(b) 3D plot: LPC characteristics.

(c) 2D plot: IPC characteristics.

(d) 3D plot: IPC characteristics.

(e) 2D plot: HPC characteristics.

(f) 3D plot: HPC characteristics.

Fig. 3.13 The compressor characteristics plots viewed on a standard desktop monitor (a, c, e) and how they are presented to the user in Virtual Reality environment (b, d, f).

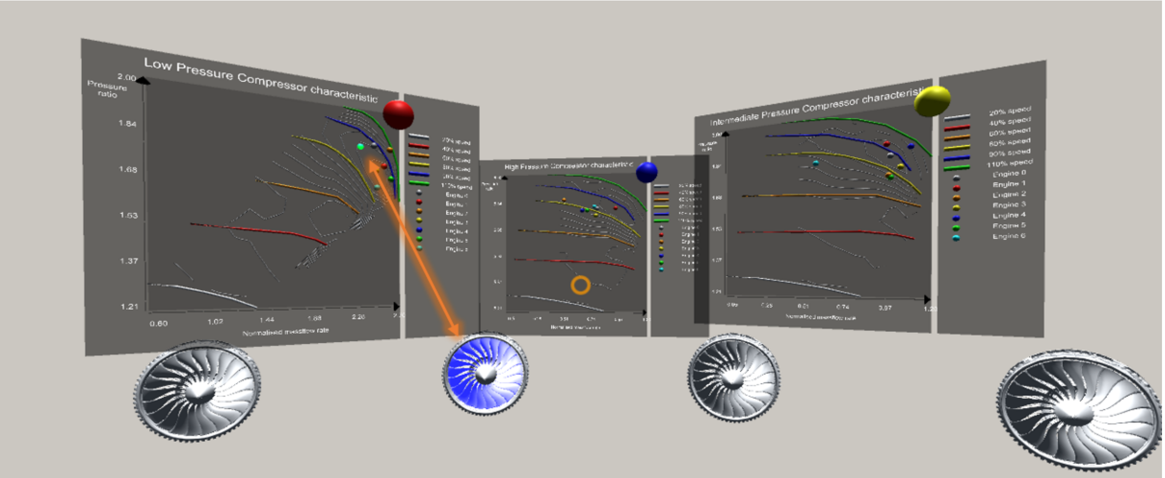


Fig. 3.14 The three graphs represent the current conditions of the three types of compressor: the low (LPC), intermediate (IPC) and high pressure (HPC) compressors for each engine. These three plots, from left to right, are shown in detail in Fig. 3.13(b), Fig. 3.13(d) and Fig. 3.13(f) respectively. Moreover, the selection of one of the LPC's operating points automatically selects the accompanying LPC component on the associated engine model as indicated by the orange arrow. The same is true for all of the other compressor characteristics graphs as well.

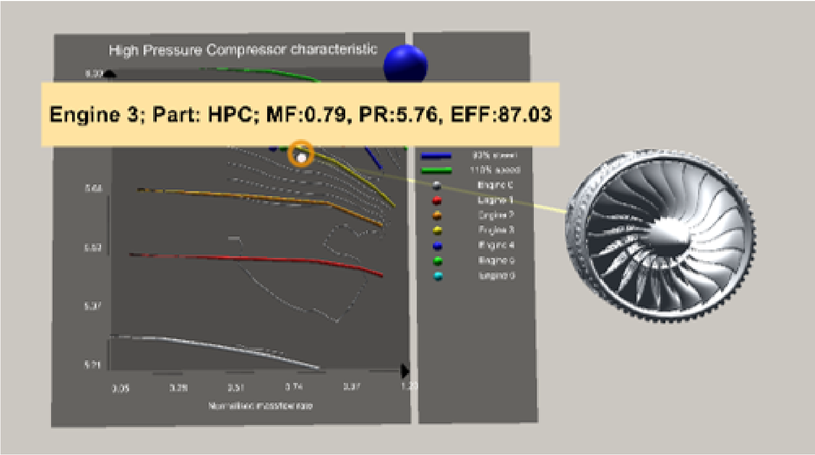

Engine 3; Part: HPC; MF:0.79, PR:5.76, EFF:87.03

Fig. 3.15 A text box presenting additional information is displayed in response to the user gazing on one of the operating points on the HPC graph. Further, a line extending from the gaze-selected point connects it with the correlated engine.

shows them as they would be seen by someone using a standard 2D screen. Here the pressure ratio is shown via a series of *constant speed* coloured lines while the efficiency is depicted via grey iso-contours. The operating point of each engine on these characteristics is shown via a series of circular markers. The second column shows the same plots as they are depicted in the VR environment. Note that the iso-contour lines and the lines are in 3D (i.e. not flat). The operating points of each engine, marked as a set of small, colour-coded spheres embedded in the plots, are interactive and if the user gazes on them the corresponding sphere displays additional information in the form of a text box: the engine identifier, the subsystem's name, along with its inlet massflow (MF), pressure ratio (PR) and efficiency (EFF), and renders a colour-coded line connecting the point with its corresponding aeroengine, as can be seen in Fig. 3.15. Legends in these three plots are set on the right side, so as not to occlude the information presented on the graphs.

The graphs are rendered semitransparent to mitigate possible occlusions effects [137]. They can be toggled to be either within the user's field of view or completely removed from it, with a press of a button on the left-hand menu, as can be seen in Fig. 3.16 (a).

Although the data shown in these graphs is not representative of the performance of real engines, the choice of the performance metrics is. Our high-level aspiration through the pipeline detailed in here, is to offer engineers a unique platform to visualise and make well-informed decisions based on the real-time working state of a fleet of digital twins.

Interaction Methods

The interaction methods use both gaze and hand-tracking mechanisms. We use the gaze-tracking part of the *Unity VR Free Sample Pack* [166]. As it is simulated by estimating the gaze position and direction roughly in the centre of the user's field of view, it does not require additional hardware. Gaze-tracking is used to inform the system which interactive object the user potentially wants to interact with. In turn, the object considered automatically highlights itself to signal to the user that it can be interacted with.

The user's gaze direction is simulated with an orange cross-hair as shown in Fig. 3.16 (a). Bimanual gestural input is used to select and manipulate the interactive objects, such as the compressor characteristics plots (see Fig. 3.20) or the aeroengines models (see Fig. 3.12). All of these objects can be rotated along a chosen virtual axis, moved, and magnified or diminished in size using a set of simple hand-gestures as shown in Fig. 3.16. Some studies conducted using the Leap Motion sensor [84], suggests that having the possibility of using virtual hand representations can aid the user's in planning and execution of complex, multilevel actions [30]. We further conjecture that virtual hand representations allow the user to better assess object size. Prior work [141] has remarked that combining the gestural input with gaze tracking can enhance object manipulation task performance.

Interaction Modes

There are two main interaction modes, these are (1) the [*engine*] mode in which the user work with the assembled engine model that is treated as a whole, compacted, object and (2) the disassembly, that is, the [*gear*] mode, switched to by pressing the [*gear*] button on the left-hand menu in which the user deals with the engine parts independently (see Fig. 3.16 (a)). If in the [*gear*] mode (see Fig. 3.16 (a)), and the user selects a data point on the plot, it will highlight (i.e. turn light-green) and automatically select the corresponding engine's component—in this case one of the compressors (see Fig. 3.21), as presented in Fig. 3.14 (a). If the same operation is invoked while the user operates in the [*engine*] mode (see Fig. 3.16 (a)), then the corresponding points across all graphs will highlight (i.e. turn light-green) and the corresponding engine will be selected as well. Furthermore, if a part of the engine is selected, the associated point on the graphs will be selected as well. Hence, this cross-selection mechanism implements the well-known visualisation technique called *linking & brushing*.

3.2.2 Object Manipulation Methods

As alluded to previously, the interactive objects, mainly the aeroengine CAD models and the compressors characteristics, can be directly manipulated by the user's hands. To support manipulation of the graphs we use the notion of *select-to-trigger-function objects* in which to manipulate the graph the user has to select it by activating one of its elements, for example, using the large sphere selector located in the top right corner of the plot. The user can also use gaze to select one of the markers placed on the graphs. The aeroengines models, and the plots as a whole, are examples of the *select-to-interact objects* technique.

Gestural Input

The system recognises four different gestures. The user's hands are tracked using the Leap Motion sensor [84] attached to the front of the headset. The virtual representations of the hands (see Fig. 3.16), are shown in the user's field of view once they are present in the sensor's detection region. Gesture recognition was built using the Leap Motion SDK [84].

Å prior study [71] suggests that using the *grasp* gesture instead of the *pinch* gesture may be more beneficial as it was favoured by the study participants. However, we found that once

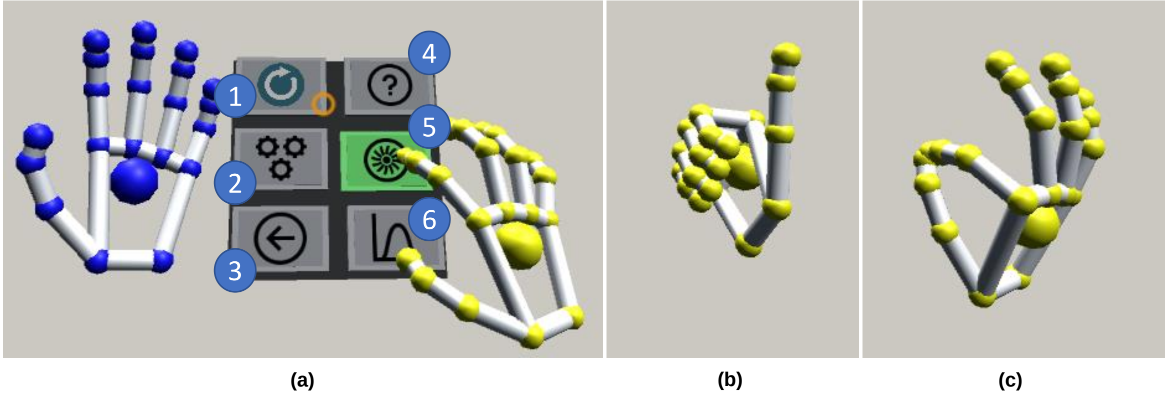


Fig. 3.16 The menu (a) opens up when the user makes the left-hand *palm-up gesture*. Then, by using the *pointing-finger gesture*, the user can invoke a series of actions: (1) double-press [*reset*] to restart the visualisation to its default settings; (2) single-press [*gears*] to be able to take the engine model apart; (3) double-press [*undo*] to reset the selected object to its default state; (4) single-press [*help*] to show a brief description of the available options; (5) single-press [*engine*] to manipulate a whole engine; (6) single-press [*graph*] to toggle the graphs. The *thumb-up gesture* (b) can be used to deselect previously selected interactive objects. The *pinch gesture* (c) is used to select (double-pinch) and manipulate objects (grasp & hold). Icons by *Icons8* (https://icons8.com).

users familiarised themselves with the pinch gesture, they were capable of manipulating the objects with ease.

There are four gestures that, depending on the mode in which the user was working in, invoke different actions as shown in Fig. 3.16. These gestures are enumerated below.

(A) **Left-hand palms-up** invokes the menu attached to the user's left-hand palm.

(B) **Pointing finger** used to single or double-press a button on the left-hand menu. The double pressing is required in case of the [*reset*] and [*undo*] buttons that once selected, invoke actions that cannot be reversed. Thus, the double-pressing requirement reassures that these buttons are not selected by mistake.

(C) **Thumb-up** is used to release the user's handle over an object. If the user would like to manipulate another object and one is already selected, the user has to articulate this gesture in order to be able to select the desired object.

(D) **Pinch** comes in two forms. First, to select an interactive object, the user gazes over it with a cross-hair and double-pinches it with either of the hands. The single-pinch gesture is used to manipulate the object.

Left-Hand Menu

The menu attached to the user's left-hand, will appear when the user turns the left-hand palm up as shown in Fig.3.16 (a). The size and layout of the menu in terms of the amount of buttons horizontally and vertically, loosely follows the design guidelines prescribed in [8], who study the feasibility of using a menu attached to one of the user's palms. As the menu itself can be recalled on demand, it will not unnecessarily occlude the user's field of view. With the help of this menu, the user can, for instance, change the mode of operation, which in turn allows for new functionality to be associated with the same, small and constant set of basic gestures (see Fig.3.16). There are six buttons in total, split into two columns with three elements in each, as can be seen in Fig.3.16 (a). These are:

1. [*reset*] double-pressed resets the visualisation to its default state as observed by the user when starting the visualisation for the first time.
2. [*gear*] single-pressed allows the user to individually interact with the sub-components of the engine models.
3. [*undo*] double-pressed restores the selected object to its default state, for example, automatically assembles and moves the previously dissembled engine into its default state and position.
4. [*help*] single-pressed reveals a text box containing user instructions.
5. [*engine*] single-pressed allows the user to manipulate the whole engine as if it would be a compact object.
6. [*graph*] single-pressed toggles the compressors characteristics graphs.

Aeroengine Models Manipulation

While in the [*engine*] mode, after selecting the model by gazing over it with a cross-hair and making a double-pinch gesture, there are three ways in which the user can manipulate the engine model. Moreover, if in the [*gear*] mode, the user can dissemble the model and manipulate each sub-component the same way as the whole model, that is, rotate, resize and move it.

1. **Rotation**: The user can, by simultaneously using both hands to grasp (i.e. by using the pinch gesture) and rotate the model in 3D space along a chosen virtual axis. The engine rotation can be seen in Fig. 3.17.

2. **Resizing**: The user can, by simultaneously using both hands, grasp (i.e. by using the pinch gesture) the object and spread their hands apart or move them closer together to increase or decrease the models' sizes respectively, as shown in Fig. 3.18.

3. **Displacement**: The user can use one of the hands to grasp (i.e. by using the pinch gesture) and move the model into any position in 3D space.

4. **Disassembly**: When switched to the disassembly mode by pressing the [*gear*] button on the left-hand menu as shown in Fig. 3.16 (a), the user can pull the model apart to inspect all of its eleven subsystems. To do so, the user gazes on and grasps the parts one-by-one as if manipulating the whole engine model. For instance, in Fig. 3.19, the user has selected and pulled apart the outer casing of the engine, exposing all the different blade rows.

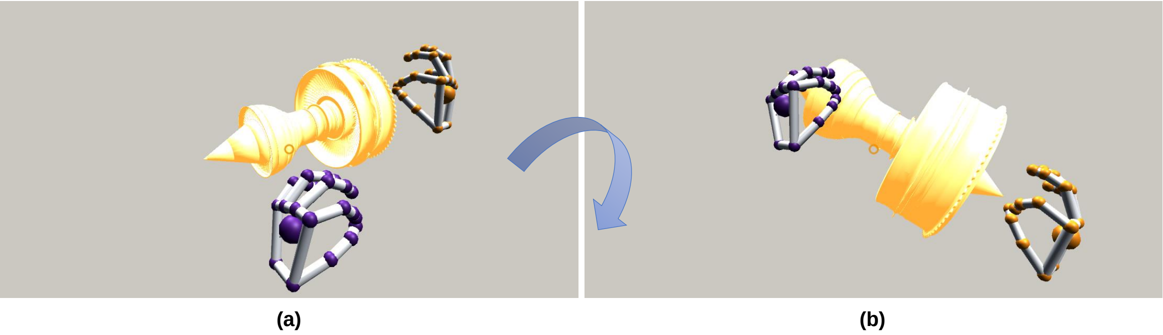


Fig. 3.17 The aeroengine CAD model rotated with the user's hand. While simultaneously making the pinch gesture with both hands and rotating one or both hands, the user can rotate the model around a chosen virtual axis.

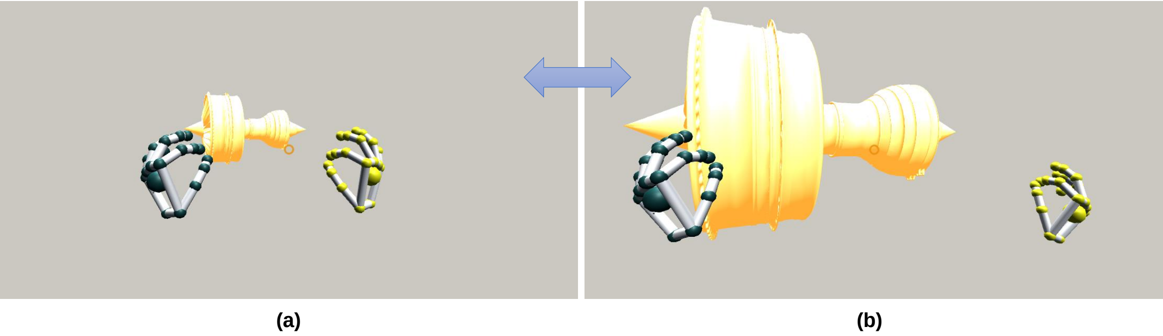


Fig. 3.18 The aeroengine CAD model magnified with the user's hands. While simultaneously making the pinch gesture with both hands and spreading them apart the user can increase the size of the model. When the user moves the hands closer the model will decrease in size.

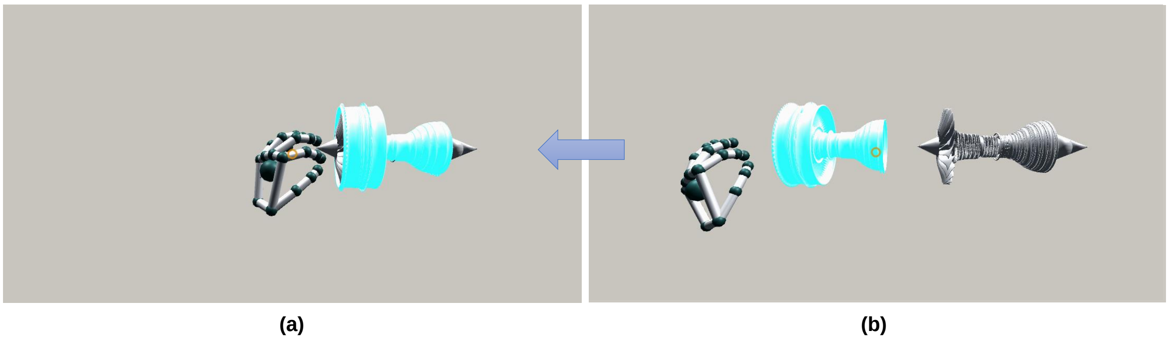


Fig. 3.19 When in the [*gear*] mode, users can interact with the specific part from engine. This figure shows the user moving the outer casing away from an engine.

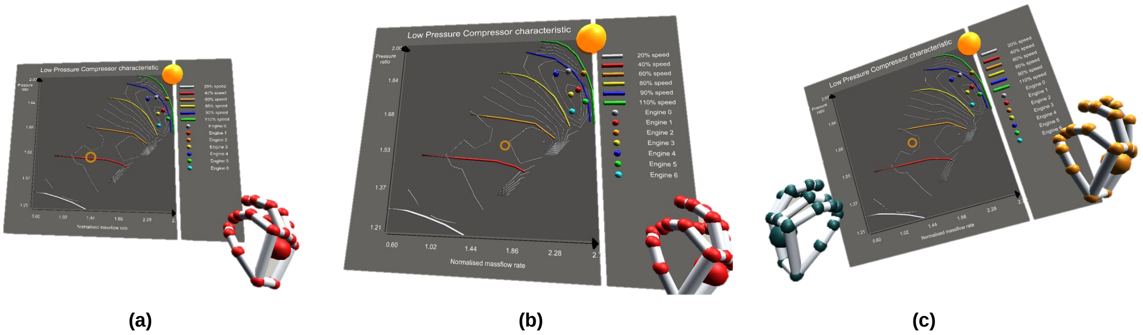


Fig. 3.20 The user can move the graph (a–b) by activating its selection controller (orange sphere in the top right corner) and simultaneously grasping with a pinch gesture and moving the graph with any of the user's hands. The user can also rotate the activated graph (c) by making a pinch gesture and rotating the graph with both hands.

Plots Manipulation

The compressors characteristics can be manipulated in almost the same way as the engine models, that is, they can be (1) rotated, (2) re-sized or (3) displaced using the same interaction methods as in case of the aeroengines. However, prior to being able to do so, the user has to gaze on a selector (large, interactive sphere placed in the top-right corner of each plot; see Fig. 3.20) and make a double-pinch gesture to select it. This feature was incorporated because the graphs themselves are composed of multiple, smaller, interactive elements, that is, the points of operations of the engine's compressors (see Fig. 3.13 (e-f) and Fig. 3.20). This allows the user to differentiate between inspecting and working with those points or the entire graph. Examples of manipulation of the plots can be seen in Fig. 3.20.

3.2.3 Interface Evaluation

We carried out a small observational study with three domain-expert participants to distill further design insights from actual use of the system. This method is listed in the survey [81] of different techniques used to evaluate interactive visualisations in the research community and is suitable when domain expertise has considerable influence on effective use of the system, which precludes the recruitment of a large number of participants for a traditional A/B evaluation.

Study Participants

The system is designed for an expert audience that in their daily work are interacting with a very specialised, bespoke environment. As such, to test our system, we recruited a group of three expert-participants, hereafter referred to as P1–P3.

Participant 1 (P1): was a final year PhD student looking at the impact of leading-edge instrumentation in turbomachinery with Computational Fluid Dynamics (CFD). He holds an MEng degree in aerospace engineering where he focused on urban pollutant dispersion through wind tunnel modelling. He reported having limited exposure to hand-tracking systems and VR.

Participant 2 (P2): was a PhD student conducting research on the low pressure compressor (fan) interactions with the intake. He holds an MEng degrees in aeronautics. He reported to have limited experience with VR and some exposure to gestural input.

Participant 3 (P3): holds an MEng degree in aerospace engineering and a PhD in Computational Fluid Dynamics (CFD) of compressors. He also worked for a year on helicopter fuselage aerodynamics, and six months on missile aerodynamics. He had no prior exposure to VR or hand-tracking systems.

Study Design

The study was carried out in two parts. First, the participants were all together shown a brief video of the system interaction capabilities. They were also given time to ask the questions and instructed to immediately stop the experiment if they felt any discomfort. Second, they were one-by-one asked to participate in the experimental phase of this study.

In the experimental phase participants were first asked to fill in the simulation sickness questionnaire (SSQ) [73] and then asked to carry out, in VR, a list of tasks. The tasks were delivered to them orally when the previous task was deemed to be finished to the satisfaction of the researcher who was always present when the participant was wearing the VR headset. Moreover, the participants were asked to think aloud and constantly discuss with the researcher what they were trying to do and what obstacles they were facing while doing it. They could also seek advice from the researcher if they did not know how to proceed further. In addition, the researcher would aid the participant if it was clear that they were stuck as the researchers could observe what the participants were seeing on an accompanying computer screen.

Once all the tasks were finished, the participants were asked to fill in additional questionnaires and to comment on, and discuss with the researchers, their experience while using the system. These questionnaires were the Flow Short Scale (FSS) [112], NASA Task Load Index (NASA TLX) [53] and Igroup Presence Questionnaire (IPQ) [61].

Tasks

The group of tasks were designed in such a way that by carrying them out, the participants would have to use the full range of interaction techniques afforded by the system. They were also based on the authors' knowledge of which tasks would potentially be interesting to the domain-expert user. The tasks were split into three parts, based on the specific compressor subsystem that the user was interacting with; these are enumerated below.

1. Fan pressure ratio performance queries:
   (a) Identify which engine's fan is the closest to choke.
   (b) Identify which engine's fan is operating at 110% shaft speed.
2. IPC pressure ratio with geometry manipulation:
   (a) Isolate each subsystem from Engine 5. There are 11 subsystems in any engine.
   (b) Identify which engine's IPC is operating dangerously close to stall (relative).
   (c) Isolate the IPC in this engine and based on an inspection of the blades, can you determine the cause of this rather high pressure ratio? (Hint: Look at the first row).
3. HPC pressure ratio with efficiency contours:
   (a) At roughly what speed is Engine 3's HPC running at?
   (b) Based on the iso-contours of efficiency, state which engine's HPC is running at the highest efficiency (relative)?

The correct answers to these tasks require the user to closely inspect and manipulate both the engine models or the graphs. For instance, in the task 2(b), the participant is required to disassemble engine 6 and possibly manipulate the IPC component to observe that it is “physically” damaged and thus causing an increase in the pressure ratio and a decrease in the massflow rate—characteristic features of compressor stall.

Observations

These main points summarise the participants’ behaviour:

- The participants were able to interact and manipulate the interactive objects smoothly after familiarising themselves with the *pinch gesture* (see Fig. 3.16 (c)). Grasping initially caused some confusion among all participants.
- Generally, the participants had no difficulty reading the information provided on the plots.
- The participants could successfully complete (see Table 3.1) the given tasks on their own, requiring only a few spoken suggestions from the observing researchers. However, participants sometimes did not identify the most efficient method possible given the system’s functionality.

There are a range of reasons that can potentially explain why it took some time for the participants to learn how to properly make the *pinch gesture* (see Fig. 3.16 (c)). First, participants were only once exposed to a brief video explaining the interaction methods before the experimental phase commenced. The impact of this *learning effect* is likely not consistent considering the participants’ variable concentration and the duration between the video and the experiment. Second, all three participants were only vaguely familiar with gestural input systems and their inherent technical limitations. For instance, P1 often closed his fist when moving the hand away, which was sometimes recognised by the system as an execution of the *pinch gesture*. Further, the participants had to be reminded that in order for the sensor to properly track their executed gestures, their hands had to be located within the sensor tracking range which would be signalled with the virtual hand avatars present in the participants field of view (see Fig. 3.16–3.20).

Once the participants gained fluency in properly executing the *pinch gesture* they were able to easily work with the interactive objects. This is most likely due to the fact that the manipulation process, to some extent, reflects how such interaction would look like in the physical world. As the participants were asked to stand while carrying out the tasks, it was interesting to observe that all of them choose to walk in the limited space constrained by the HMD cables. This may be explained by the fact that, the engines models were spread

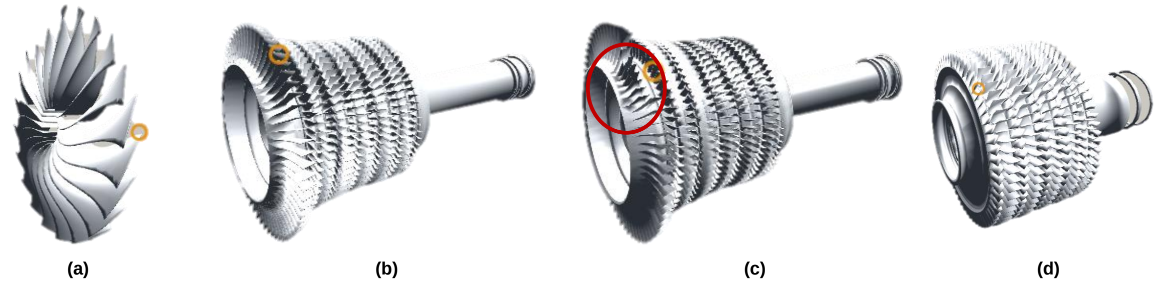


Fig. 3.21 The CAD models of the compressors: (a) low pressure compressor (LPC), (b-c) intermediate pressure compressor (IPC); (d) high pressure compressor (HPC). For the clarity, the scale of the aerengine parts is not preserved. The nominal IPC is shown in (b) along with the damaged engine (highlighted in red) in (d).

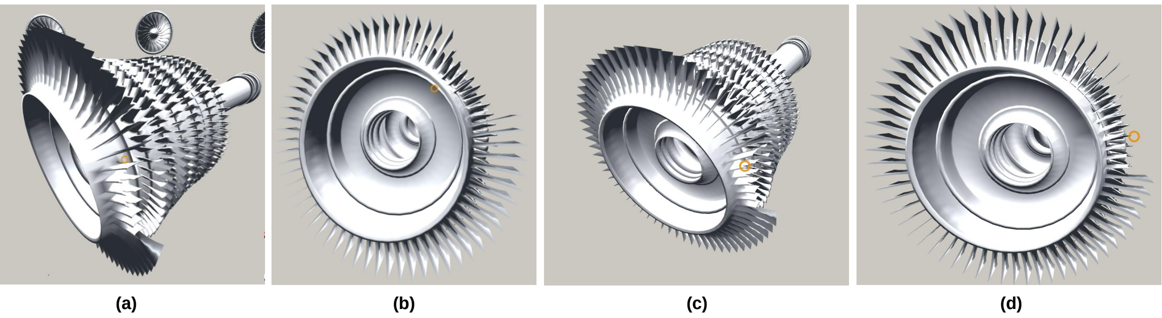


Fig. 3.22 The CAD model of the faulty intermediate pressure compressor (IPC) seen from different perspectives after being rescaled and rotated by the user.

horizontally in such a way that the participants could not see all the models and graphs at once (see Fig. 3.12 and Fig. 3.14).

During the study it was also observed that the participants seemed to be mildly confused by the relationships between the compressor characteristics and the engines (see Fig. 3.14). For instance, P1 forgot that it was possible to select a point in the graph to simultaneously select a specific part in an engine (the LPC, IPC or HPC). Instead, to locate the engine of interest, he chose to follow the direction of the lines linking the points of operation on the graphs with the corresponding engine by moving the entire plot with him, and, in steps, placed it closer to the target engine. Then, after finding the corresponding engine, he disassembled it to be able to access the part (i.e. the broken IPC in Task 2 (c) as seen in Fig. 3.21 and Fig. 3.22. After being told that he could select the part directly through the plot by switching to the [*gear*] mode (see Fig. 3.16 (a)) he believed he could have performed better if he was familiar with the technique in advance.

Furthermore, the participants did not leverage all the available features. For instance, P3 commented that he often found having the plots visible in his field of view was annoying and he would prefer to have a features allowing him to hide them on demand. In fact, this functionality did exist in the system and could be easily accessed through the left-hand menu as show in Fig. 3.16 (a).

Questionnaires Results

The results of the SSQ showed that P1 developed some slight symptoms of the simulation sickness (1/27 and 3/21 for nausea and oculo-motor strain respectively) whereas P1 and P2 experienced none.

The results of the NASA TLX (P1: 66/100; P2 66/100; and P3: 69/100) indicate that carrying out the given tasks required some effort from all three participants—a point that may be explained by the participants' low levels of prior exposure to both VR and gestural-input interaction methods. Moreover, the task partially relies on the participants domain knowledge which could also add to their cognitive load.

The results of the IPQ showed that the participants reported relatively high levels of the feeling of presence in the simulation with all the scores above 50% (P1: 3.64/7.0; P2:3.93/7.0; P3:3.57/7.0 respectively), which suggest positive engagement with the system.

The results of the FFS are given as two components: flow and anxiety levels. The participant reports are as follows. For participant P1: flow of 4.9/7 and anxiety of 2.5/7; P2: flow of 4.7/7 and anxiety of 2.5/7; and P3: flow of 3.4/7 and anxiety of 3.0/7. The relatively high levels of flow and low of anxiety across all the participants could indicate that they did not feel *out of place* when interacting with our visualisation.

Tasks Results

The results of the task should be interpreted with caution as the participants were given unlimited time to finish the task and frequently sought the researchers help if they got stuck. Hence, even though the researchers put an emphasis on not leading the participant to answers, the answers are subject to bias. Regardless, it is worth noting that all participants managed to carry out the tasks correctly, with the exception of P1, who incorrectly identified engine 4 instead of engine 5 when using the LPC compressor characteristics plots. P1 remarked that he had no familiarity with the low pressure compressors. The correct answers to all the tasks are given below:

1. Fan pressure ratio performance queries:

    (a) The fan operating closest to choke is the fan on engine 5.

    (b) The fan on engine 2 is operating at 110% shaft speed.

Table 3.1 The table shows the tasks results for each of the three participants P1–P3. The cells with correct answers are marked with a light-green background whereas the only incorrect answer is marked with red background (i.e. answer to Task 1 (a) given by P1). Whenever the task required visual inspection that the researcher had to deem complete, i.e. 2 (a) and 2 (c), the table indicates whether it was carried out with satisfying results (correct) or not (incorrect).

Task	P1	P2	P3
1(a)	engine 4	engine 5	engine 5
1(b)	engine's 2 LPC	engine's 2 LPC	engine's 2 LPC
2(a)	correct	correct	correct
2(b)	engine's 6 ICP	engine's 6 ICP	engine's 6 ICP
2(c)	correct	correct	correct
3(a)	80% speed	80% speed	80% speed
3(b)	engine's 1 HPC	engine's 1 HPC	engine's 1 HPC

2. IPC pressure ratio with geometry manipulation:

   (a) There are 11 subsystems in any engine as can be counted on the engine CAD model.

   (b) The IPC on engine 6 is operating dangerously close to stall.

   (c) The blades in the first row on the engine 6's IPC have been damaged (see Fig. 3.21 (c)).

3. HPC pressure ratio with efficiency contours:

   (a) Engine 3's HPC is running at roughly 80% speed.

   (b) Based on the iso-contours of efficiency, engine 1's HPC is running at the highest efficiency (relative).

3.2.4 Limitations

Some limitations of the system functionality were imposed by the available data, system design and the hardware. For instance, the VR headset was connected to the supporting PC with USB and HDMI cables. In addition, the Leap Motion required separate USB connection. These cables restricted the range of participants' motion, as care must be taken to ensure that participants do not stumble or entangle themselves with the cables.

The participants chosen in our study were not pre-screened for symptoms of the colour vision deficiency. In our system, being able to differentiate the colours is quite important, as we used the colour-coding for both the selected engines as well as to colour-code the data plotted on the graphs. In this respect, P1 mentioned that when the engine was highlighted, he could not clearly see the damaged blades on the IPC compressor in engine 6 without looking closely at the model.

3.2.5 Design Implications

Here, we present a list of tentative design suggestions for designing of a VR system that use gestural input coupled with gaze tracking to interact with a complex, multi-element immersive visualisation in VR. This list is guided by our own observations of the participants' behaviour while wearing the VR HMD as well as the participants' comments.

DI1—Training: Due to the novelty of the hand-tracking, as well as VR headsets, these technologies are not yet widely adapted in either mass-market, industry or academia. Hence, as observed in our study, the users must be provided with training in how to interact with the system using gestural input.

DI2—Object Placement: In general, placing objects, in this case the aeroengine models (see Fig. 3.12), around the user in the horizontal plane was potentially better than using the vertical plane. However, to aid the users in finding objects outside of their field of view, additional navigational indicators may need to be developed and tested. P2 suggested that some form of a conveyor belt that would rotate the engines in his field of view might be helpful, instead of having to repositioning himself in 3D space. P3 pointed out the separation between engines and graphs may cause confusion to participants. He also suggested that only after an engine or a part is selected, should the plots become visible.

DI3—Complex Graphs: Following the suggestion in [42] and the comments of P3, the use of 3D graphs could be better utilised. For instance, the iso-contour plots, such as the ones used in our study, could potentially be more beneficial to the users if the contour-lines would extend from the plot, forming a 3D-like structure. Then, given the plots' rotation and re-sizing capabilities, it would be easy for the user to spot anomalies and outliers.

DI4—Object Manipulation: Prior related research [30], suggests that bimanual manipulation of objects is a very promising and favourable way of interacting with CAD and other types of 3D models embedded in a VR environment. In practice, the gaze tracking system caused some trouble for P1 when trying to observe details of the damaged engine as the object had reduced contrast when highlighted by the gaze cross-hair. P1 suggested a feature to deactivate the gaze tracking-based highlighting.

3.2.6 Discussion

Given the limitations of both hardware and the study design, to some extent imposed by the difficulty in recruiting a large number domain-expert participants, the results of the experimental study should be interpreted with caution. This small formative study was designed to help us identify initial usability problems, spot potential improvements, and gain insights into aerospace VR design interaction capabilities. It does, however, allow us to trim the vast design space and to guide further design efforts in this area.

One of the most interesting elements of the visualisation was the possibility of disassembling the CAD aeroengine model into its composite subsystems, which could be independently manipulated and inspected in the same way as the entire engine. Coupling these with the compressor characteristic plots allowed the participants to carry out a complex, multilevel tasks in VR. Although not completely satisfied with our system as stated in the remarks made by the participants, they all managed to correctly finish the given tasks. We believe, that to some extent, their dissatisfaction may stem from the short duration of the training provided in a form of video screened before the experiment commenced. Most likely, if given a greater familiarisation period whilst immersed in the VR environment, for example, by providing a prior training phase, the participants would be able to carry out the task more favourably, leveraging all the functionality that the system has to offer, such as, an ability to turn on and off the plots on demand. Also, as mentioned previously, the participants could potentially quickly accustom themselves with the *pinch gesture* which in turn would allow them rapidly grasp and manipulate objects.

3.3 Conclusion

The goal in the Section 3.1 was to introduce the *AeroVR* system—a novel VR aerospace design environment with a particular emphasis on dimension reduction. This work identified the main structures of the design environment and implemented a fully working system for commodity VR headsets. This interface was also verified from two perspectives: usability and expressiveness.

The two main identified tasks were (i) gaining the overview over the design and (ii) comparing the nominal geometry with the one associated with a specific performance parameters. The former was achieved through a mixture or visualisation (e.g., blade and engine geometry, and scatter plots) and lower-level tasks (e.g., movement and interaction). The latter was achieved through visualisation of the overlaying geometries: semi-transparent nominal blade superimposing over solid shape of a new design.

Section 3.2 introduces an interactive, immersive system for the simultaneous visualisation of the performance of a fleet of aeroengines in VR. Presented first, an earlier iteration of the system was vastly extended and equipped with a range of new capabilities. The version discussed in Section 3.2 uses 3D geometric computer-aided design (CAD) models of the engines paired with compressor characteristics maps plotting the pressure ratio and efficiency as a function of shaft speed and inlet flow capacity of the low (LPC), intermediate (IPC) and high pressure compressors (HPC) for each engine. Superimposed on these maps are the true performance of each engine, obtained through real-time sensors. This allows the user to promptly diagnose and spot potential issues with the engines across the entire fleet.

As discussed earlier, the selection of an appropriate interaction technique is one of the key ingredients of such a system. Here, the work outlined in this Chapter explored how can we leverage interaction methods facilitated by gaze-tracking coupled with hand-trackingwith no need for the user to hold a controller. Here, an observational study with a small number of domain-experts to distill a list of design implications relating to the development of the VR visual analytics platform.

With respect to the prior work outlined in the literature review in Chapter 2, similarly to some of the earlier studies, [85, 181], the AeroVR system described in Section 3.2 do combine CAD models together with plots visualisation. However, it focuses on using a new, bimanual input facilitated by a hand-tracking and gesture-recognition. Moreover, to date examples of aerospace design [67] were not studied in the context of an immersive VR-based environment.

Chapter 4

Virtual Reality Analytics in Civil Engineering

Photogrammetric techniques offer a promising method for acquiring virtual high-quality content data and can be used to generate photo-realistic 3D models of physical objects and structures. Such methods may rely solely on camera-captured photographs as its input or may also include additional sensor data. In addition, photogrammetry is a promising approach for the generation of 3D models used to prepare digital twins of real-life objects and structures. When of sufficient quality, digital twins can prove an effective archival representation and may offer a substitute for site visits and inspections. While photogrammetric techniques are well-established, insights about effective methods for interacting with such models in virtual reality remain underexplored.

This Chapter reports the results of a qualitative engineering case study in which we asked six domain experts to carry out engineering measurement tasks in an immersive environment using bimanual gestural-input coupled with gaze-tracking. The qualitative case study revealed that gaze-supported bimanual interaction of photogrammetric 3D models is a promising modality for domain experts. It allows domain experts to efficiently manipulate and measure elements of the 3D model. To better allow designers to support this modality, this Chapter reports design implications distilled from the feedback from the domain experts.

4.1 Conducting Engineering Surveys in Virtual Reality

A major obstacle towards the mass adaptation of Virtual Reality (VR) is a lack of high-quality content. Even though computer-aided design (CAD) models can be visualised in VR, in the majority of cases such models will still have to be manually prepared, which is

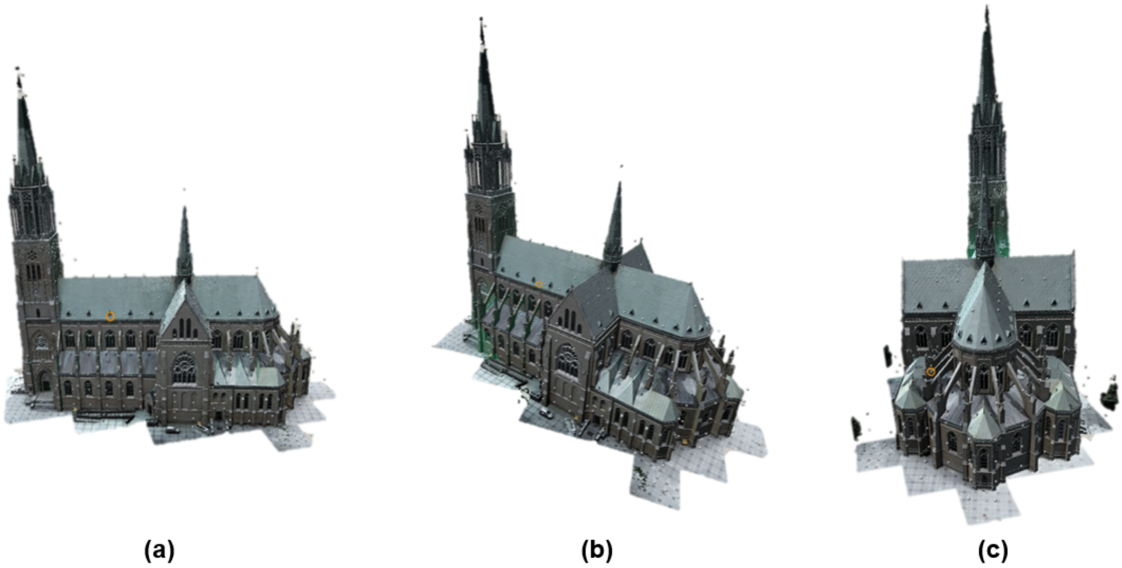


Fig. 4.1 The 3D photogrammetric model of the Archcathedral Basilica of St. Stanislaus Kostka in Łódź, Poland: (a) side view; (b) three-quarter view; (c) rear view.

a resource-intensive task. Further, in order to generate increasingly realistic models that resemble real-life objects and structures, there is a need to apply high-quality textures to the models. This is particularly important if there is a desire to enhance the immersiveness experienced by the users [167].

One of the most promising approaches for semi-automatic generation of such high-quality 3D models is *photogrammetry* [162]. This approach uses 2D information, such as photographs of an object, that can then be combined with additional data gathered by a wide array of various sensors, such as GPS location data, to reconstruct digital photo-realistic 3D models of real-life objects [4].

However, photogrammetric models may suffer from certain deficiencies. For instance, due to insufficient or corrupted input data, the process of generating such models may introduce “sharp edges”, “blank spaces” or break the model into multiple disconnected parts. For instance, see the tip of the archicathedral tower in Fig. 4.1. This in turn may affect the way in which the user would like to interact with the data.

Nevertheless, photo-realistic models generated from real-world content can be used to gather additional information about the as-it-is reflection of the current condition of the modeled object. Here, depending on the model and its purpose, it may be feasible to, for instance, measure the various dimensions of features of such models without the need to actually measure the existing object in physical space [1, 155], which is often expensive and cumbersome in practice.

Moreover, the capability of capturing the real-world objects and structures in the as-they-are state shows the great promise of the photogrammetric methods for the purpose of digital twinning. If such 3D models are of sufficient quality, digital twins can be used as an effective archival representation and may offer a substitute for site visits and inspections of heritage [6, 127], engineering, manufacturing or industrial sites [35] as well as in asset management [91]. Saddik et al. remarks that in order for unlock the full potential of the digital twins, a number of key technologies have to be utilized, listing among others, the Virtual and Augmented Realities [35].

In recent years, the VR community has successfully used photogrammetry to augment and enhanced the fidelity of the user's experience while using an immersive interface, see, for example, Vajak et al. [167]. Moreover, Antlej et al. [6] noted that when acquiring data for their system, models prepared using photogrammetry offered satisfactory results for VR applications. Importantly, there is only a little prior work that has focused on interacting with the generated 3D models in VR by using virtual hands, using unencumbered fully tracked gestural input. For the work regarding non-photogrammetric object manipulation in VR using mid-air interaction, see for instance [92, 17] whereas an interesting method for the bimanual 3D virtual object manipulation can be found at Song et al. [144]. However, the literature does indicate there is potential with such approaches. For example, a usability study conducted by See et al. [127] concluded that using even low-fidelity virtual hands facilitated by hand-held controllers were better than using virtual device avatars to represent controllers.

In this work we investigated gaze-supported bimanual interaction to allow domain experts to manipulate a complex photogrammetric 3D structure model and take distance, surface and volume engineering measurements on it. Various measurements of the as-they-are objects are common and important in many application areas, such as surveying, computer-aided design (CAD), structural engineering and architecture. Gaze-tracking and head-tracking are used as means to select a given interactive item. In this research we estimate gaze-tracking using head-tracking. It is not as precise as eye-tracking as the direction of the user's head and eyes are not necessarily always aligned with each other [140, 139]. The bimanual interaction technique has been an item of interest and studies in the human-computer interaction for many years, for instance, see Buxton et al. [16], Guiard [50] or Balakrishnan et al. [11]. In the latter paper, the authors reported that in their experiments involving carrying out tasks in a 3D virtual scene, the participants had strongly favored the bimanual technique over the unimanual interaction [11].

We report on an engineering case study with six domain experts to better understand how novel gaze-supported bimanual interaction techniques would satisfy domain experts' needs of wants in the task of surveying complex 3D structures, such as buildings. The immersive VR environment used in this study is discussed in more detail in Tadeja et al. [155].

In engineering contexts, CAD models are usually representing idealised and generic items that are typically used for design and evaluation. Environments to support these models must allow not only repositioning but also modification, for example, scaling, separation, and combination of individual parts. Together, these interactions may support more advanced and more complex tasks such as computer simulations.

However, 3D photogrammetric models, are often representative of specific real-world objects or structures, such as buildings, captured in their as-they-are states. Such 3D models have applications in surveying, preservation and maintenance. As such, effective manipulation and interaction techniques for 3D photogrammetric models give rise to different constraints than ordinary the CAD models.

For example, where parts of the CAD models may be freely resized, in surveying with 3D photogrammetric models, the elements of the model must not only reference the actual physical characteristics of the original object but must also be scaled together when resized. It is essential, for example, that relative dimensions are not modified as this could invalidate the survey's results. Instead, interactions of this type are limited to the viewport. This type of specialist evaluation is not concerned with the simulated properties of the model, but the recorded physical characteristics that might vary considerably from the idealised reference. Measurement tools are therefore of crucial importance for evaluating distortions in the surveyed object. Unlike simulations that are designed to support experimentation, real-world properties of photogrammetric models might highlight critical safety concerns, such as sagging beams, subsidence, or dangerous corrosion. Such information, native to photo-realistic 3D models, would have to be additionally provided to ensure the same value for the user when working with CAD models.

4.2 Minimal Design Requirements

In this section, we describe the requirements associated with the two main components of the VR system designed to conduct an engineering survey of a real-life structure. These two interlinked functions are (1) model manipulation (Fig. 4.5); and (2) model measurement (Fig. 4.7). The ability to manipulate the model is a significant, and potentially limiting, factor in the measurements the user may make. The initial functionality was based on the authors' experience as active interaction design researchers and on prior observation of professionally active engineers. Guided by observations of participants' behaviour and their suggestions, we have then extended our system with new capabilities that allow users to take surface and volume measurements of such models. This enhanced functionality necessitated the model measurement requirements to be split into three sub-components.

Here, for simplicity, we are focusing on the minimal viable cases that can be easily extended to cover more complex tasks as well. For instance, taking a distance measurement, understood as Euclidean distance in 3D space, using Cartesian coordinates requires at least two points. The length of the line connecting these two points constitutes the distance measurement. Similarly, in the case of the surface and volume measurements, consistent with common mesh-based 3D environments, we require coordinates of at least three and four points to calculate the triangle's surface area (Fig. 4.9) and pyramid volume (Fig. 4.10) respectively. These calculations have to be adjusted to take into account that the triangles and pyramid surfaces can have various inclination angles in 3D space. Since 3D objects are essentially constructed out of triangle meshes it is possible to use these basic measuring tools to measure dimensions of more complex structures.

DR1—Model manipulation: The system should provide an easy-to-learn and effective method for manipulating the 3D model so that the interaction technique does not negatively affect the user's main task of identifying and taking desired measurements of the 3D model (Fig. 4.4 and Fig. 4.5).

DR2—Model measurement: The system should allow the user to take essential engineering measurements of the model, including various distances (for example, height and width), surface and volume of its part or a model as a whole.

These measurement requirements can be further decomposed into the following three sub-components:

DR2D—Distance measurement: Taking the distance measurements should allow the user to effectively measure any distances (in chosen unit of length, such as meters [m]) between any two chosen points in the 3D space be it on, within, or near the vicinity of the model (Fig. 4.8 and Fig. 4.7).

DR2S—Surface measurement: Taking the surface measurements should allow the user to effectively measure any surface area (in a chosen unit of surface area, such as square meters [m^2]) spanning over at least three points regardless of the angle of inclination of this surface in the 3D space. Such surfaces could be localised on top, in the near vicinity, within, or crossing over the borders of the model (Fig. 4.9).

DR2V—Volume measurement: Taking the volume measurement should allow the user to effectively measure any volume (in chosen unit of volume, such as cubic meters [m^3]) given by at least four chosen points in the 3D space be they on, within, or near the vicinity of the model (Fig. 4.10).

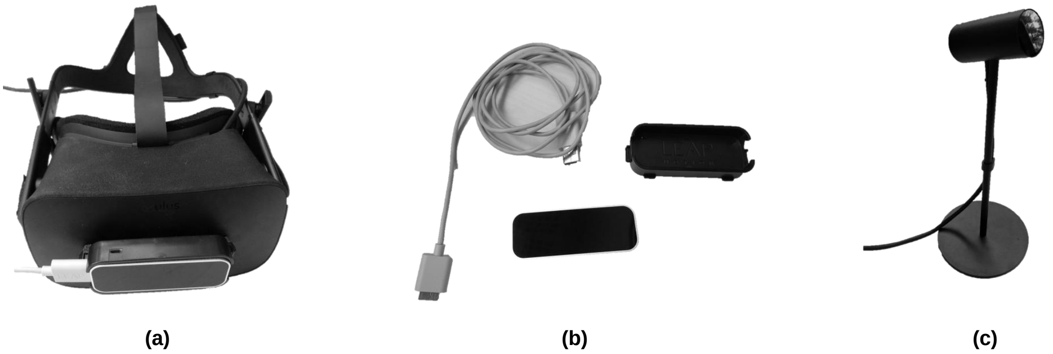


Fig. 4.2 The visualisation and tracking hardware used in our study: (a) the Oculus Rift with mounted Leap Motion; (b) Leap Motion hand-tracking sensor with mounting case and a USB cable; (c) the Oculus Rift's tracking sensor. All three components have to be connected to a supporting PC via USB (a-c) and HDMI (a) cables.

4.3 Case Study

4.3.1 Visualisation Framework

The hardware supporting our system consisted of an Intel i5-9400 CPU and NVidia GeForce GTX 1660 Ti GPU. The head-mounted display (HMD) was an Oculus Rift [105] as seen in Fig. 4.2. The hand-tracking was powered by the Leap Motion Sensor (LMS) [84]. The visualisation was written using the Unity game engine and various supporting assets, and the LMS Software Development Kit (SDK) [84].

Archcathedral Model

The 3D photogrammetric model used in the user study was a photo-realistic recreation of the Archcathedral Basilica of St. Kostka in Łódź, Poland that can be seen in Fig. 4.1[1]. The data acquisition as well as the final model were prepared by experts from the *Softdesk CS* in Łódź, Poland. The photos were captured by an unmanned aerial vehicle (UAV) and then later processed by photogrammetry software. The data acquisition process was divided into two stages: the main building and tower. There were 1754 photos taken during six independent flights. Those were conducted on different dates, which influenced the accuracy of the image metadata as the altitude measurements depend highly on atmospheric pressure, which differed between the flights. As a result, the acquired data was not properly georeferenced, which resulted in a loss of accuracy.

[1] http://softdesk.pl/katedra

Coking Plant Model

The 3D photo-realistic model of a coking plant was used to test the additional features of capturing surface and volume measurements of the 3D model in VR as seen on Fig. 4.9[2] and Fig. 4.10 respectively. The imagery data was obtained during a few UAV flights above the plant during one day and took around two-and-half hours. After aerotriangulation processed with a root mean square reprojection error of 0.49 pixels, a 3D model was reconstructed from 1491 photos. Additional sensor data gathered during UAV flights was used to georeference and scale the final model. The data acquisition as well as the final model were prepared by experts from the *Softdesk CS* in Łódź, Poland.

4.3.2 System Structure

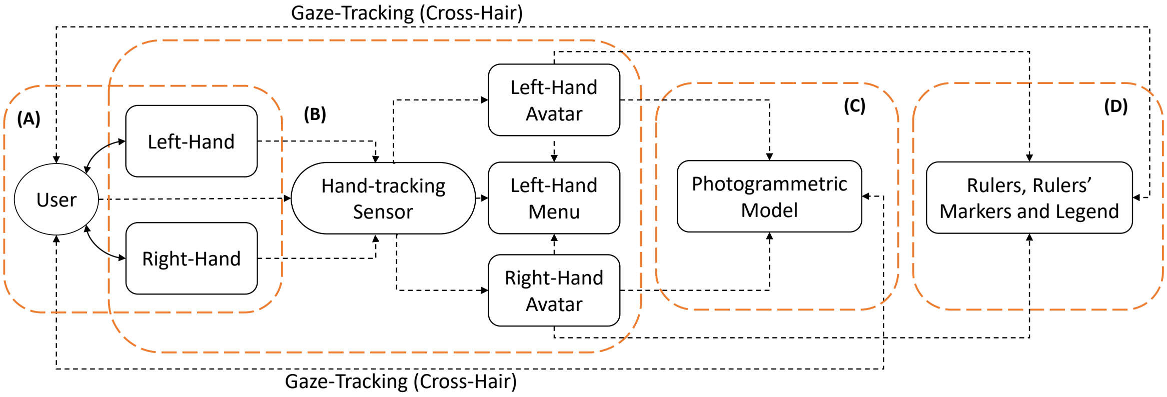


Fig. 4.3 The diagram models the signals flowing between the four main elements of the visualization system: (A) the user; (B) the hand-tracking setup; (C) the 3D model; and (D) the measurement toolkit. Both unidirectional and bidirectional signals are shown. Components of (A) and (B) overlap as the user's hands are part of both the hand-tracking (B) and the user (A). The dashed arrows mark the uni and bidirectional signals flow between the visualisation elements, while the physical connections of the user's limbs are marked in solid arrows.

We decided to observe the way signals flow within the system in order to reason about its overall structure (Fig. 4.3). We observed four key elements: (A) the user; (B) the hand-tracking sensor coupled with the corresponding virtual hand representations; (C) the photogrammetric model of an object or structure; and (D) the measurement tools.

(A) **User:** User interaction in the system is driven by two interaction styles: 1) a combination of gaze-tracking coupled with hand-tracking; and 2) gesture recognition. The user exchanges signals with the system in two ways. First, there is a bidirectional signal via

[2] http://softdesk.pl/koksownia

the system's depiction of a ray-traced orange cross-hair, which signals to the system what the user is focusing on at the moment. While the user is gazing at an interactive element (see Fig. 4.7), the system responds by highlighting such an object, thereby signalling to the user that this element can be interacted with. After object selection, the user can use a hand-gesture (see Fig. 4.4) to invoke an action dependent on both the gesture type and the object itself. Further, any kind of manipulation of a model or measurement operation is signalled to the user by immediate system responses (see Fig. 4.5).

(B) **Hand-tracking:** As the main method of interaction is gestural-input facilitated by the external sensor (see Fig. 4.2 (b)) [84], the signal received and sent by this component are the results of a constant feedback loop. The position of the hands and their gestures are always signalled to the user via virtual hand representations (see Fig. 4.5-4.7). If the virtual hands are not visible to the user then the physical hands are outside the sensor's tracking area. Actions executed by the user on either the model (C) or the measurement tools (D) are immediately applied, and, as such, signalled to the user via a range of visual clues. For example, the ruler markers are ready for interaction when a colour-coded halo appears (e.g., see Fig. 4.7)

(C) **Photogrammetric model:** The model (C) responds instantly to all operations initiated by gestural input. These operations include object rotation, resizing and displacement (see Fig. 4.5). All such manipulations are instantly reflected in the visualisation.

(D) **Measurement tools:** The measurement toolbox includes the rulers (vectors/lines in the 3D space), ruler markers (small, spherical markers) and the legend shown to the user through the heads-up display (HUD) (see Fig. 4.7 and Fig. 4.8). All these objects respond to actions invoked with the user's hands. The markers can be created, displaced, removed and highlighted while the user is gazing at them with a cross-hair. Their halo visualisations signal to the user which hand they can be moved with: blue or red for the left and right hand respectively. The halo is green when the user is in the process of connecting the markers with a ruler. The rulers themselves change their length and position depending on the user's actions and can be completely removed from the visualisation, if desired. The legend itself responds to the changes made over the rulers, revealing the most recent state of any taken measurements.

4.3.3 Snapping Grid

The 3D model is highly detailed and as a result it is impractical to expect a user to directly interact with individual points in the model. Instead, to facilitate an easy-to-understand measurement basis, we surrounded the 3D model by a snapping grid generated in the form of snapping points (represented as small spheres), positioned equidistantly around the model's mesh. Whenever the user was placing a ruler's marker in a vicinity of a snapping point, the marker would automatically snap to its position.

Further, it is possible to automatically place a snapping grid around the larger part of the model, such as its roof. It can be done relatively simply by approximation techniques using the mesh information alone. However, being able to automatically place the snapping grid around smaller features of the model remains an open problem in the literature.

4.3.4 Interaction Methods

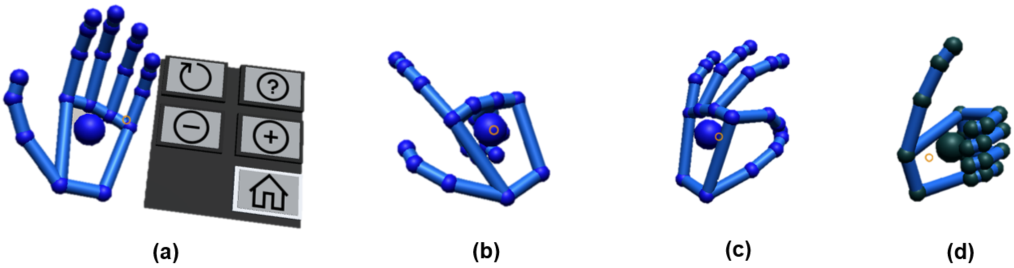


Fig. 4.4 The four gestures recognised by the hand-tracking: (a) the *left-hand palm up* calls the menu; (b) the *pointing finger* to press the menu buttons; (c) the *single-hand pinch* gesture; and (d) the *thumbs-up* gesture. Icons by *Icons8* (https://icons8.com).

The system used a mixture of interaction techniques. It combined the hand-tracking facilitated by the LMS mounted in front of the HMD (see Fig. 4.2) with gesture-recognition capabilities afforded by the LMS SDK. The system also used gaze-tracking, which was built upon the *Unity VR Samples Pack* and used a cross-hair metaphor and ray-tracing to approximately estimate which object the user was gazing at.

As reflected by Slambekova et al. [141], combining gaze and gesture may positively influence the interaction results as more information about the user's intent is communicated to the system. Since the user can take as many measurements as desired, there may be a substantial number of interactive objects (for instance, the ruler's markers) present in the 3D scene, which were previously generated by the user. Without the use of gaze-tracking, we would have to apply a different, more complex mechanism to allow the user to swiftly select

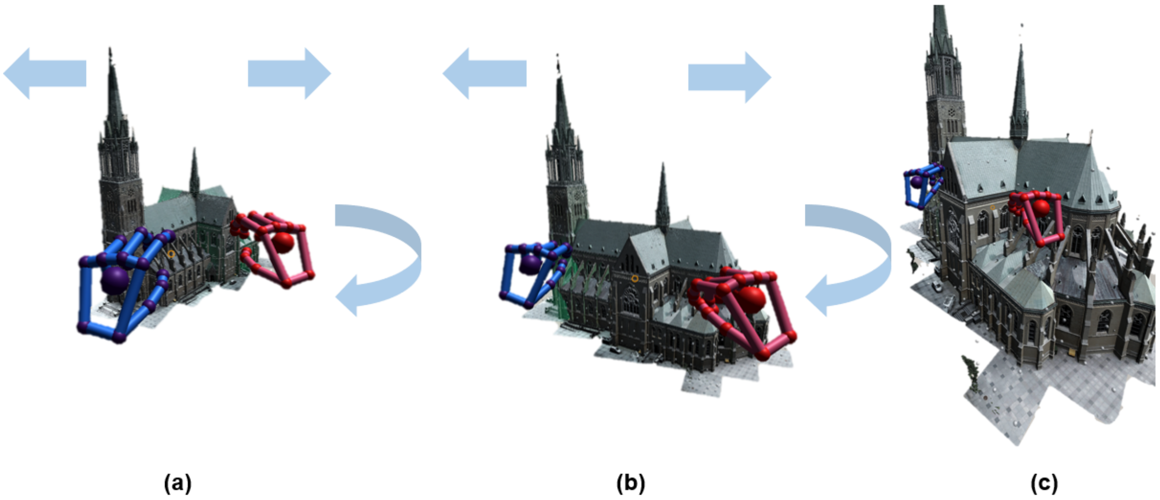


Fig. 4.5 Manipulation of an object: the user rotates the model by making a pinch gesture with both hands and moving one hand (in red) in an arc whilst the other hand serves as the rotation axis (in blue). Simultaneously, by moving the hands apart, the user enlarges the model.

between these objects. For example, the user would have to use their hands to first grab an object. This action could cause an unintentional shift of the ruler's marker prior to taking a distance measurement by creating a spurious connection with another marker. To remove this risk, we would have to create another operation mode invoked either through a new hand gesture, inclusion of a new button, using the left-hand menu (Fig. 4.4), or a combination of both. Another option would be to implement raycasting selection that extend a ray pointing towards the selected interactive object. Again, in such a case we would have to couple the selection with an additional gesture made with the user's other hand. In both cases the user's gaze would most likely be directed towards the near vicinity of the object the user desires to interact with. Hence, using gaze-tracking to assist in the object's selection procedure is straightforward.

The system recognised four main gestures (illustrated in Fig. 4.4): (a) the *left-hand palms up* gesture which called up the menu; (b) the *pointing finger* gesture which was used to press a button on the menu; (c) the *pinch* gesture which was used to drive all primary interaction; and (d) the *thumbs-up* gesture which was used to release the hold on an object. The size and placement of the menu loosely followed the design guidelines distilled by Azai et al. [8].

4.3.5 Object Manipulation Methods

The system provided the user with three main methods for manipulating the 3D model, illustrated in Fig. 4.5. These relied mainly on the double or single-handed pinch gesture

(Fig. 4.4 and the LMS SDK documentation for more details) and allowed the user to: (1) move and reposition the model in the 3D space; (2) decrease or increase the model's size; and (3) rotate the model in the *X*-*Z* plane.

4.3.6 Measurement Method

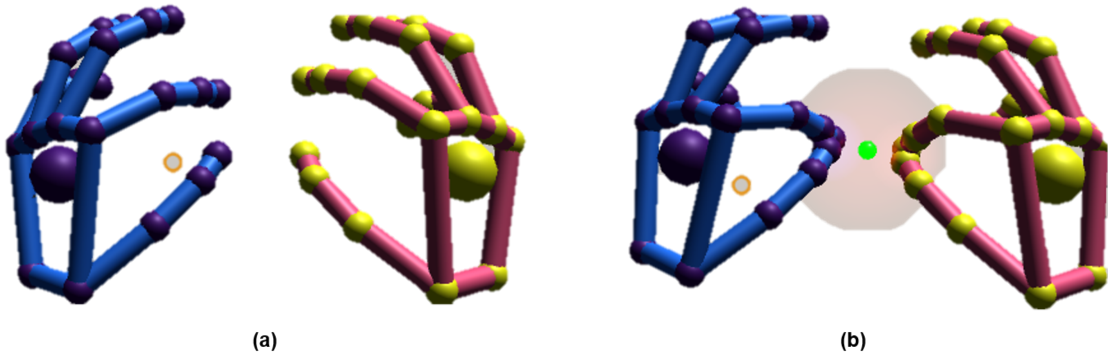


Fig. 4.6 To generate the new rulers' markers, the user has to make a pinching gesture with both hands (a) with the all four pinching fingers at once i.e. the thumbs and pointing fingers close to each other. Then, a new marker will be generated between the user's hands (b).

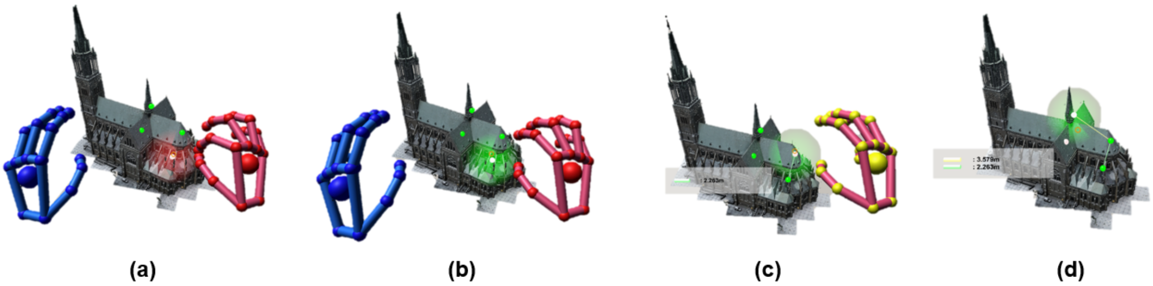


Fig. 4.7 The user's field of view (a–c) as the user is taking measurements of the model carried out with help of our system: (a) placing the ruler's markers; (b) selecting the starting marker to be connected with a ruler with another marker; (c) selecting a previously connected marker to be a starting point of another ruler.

The user could acquire the model's dimension measurements by generating (see Fig. 4.6) and placing rulers' markers around the object's mesh. Each pair of those markers could then be connected with a ruler, that is, a 3D vector where the magnitude was the measurement, revealed on the head-up display (HUD; Fig. 4.8).

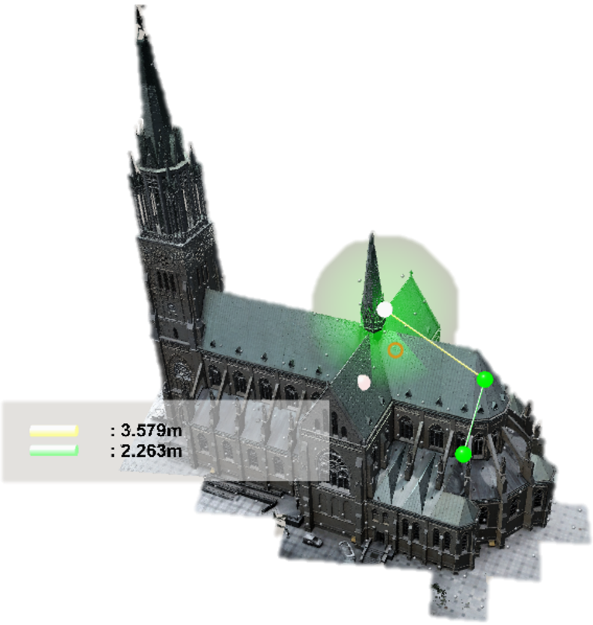


Fig. 4.8 Example of distance measurements of a photo-realistic recreation model of the Archcathedral Basilica of St. Kostka in Łódź, Poland.

4.4 Engineering Surveys in Real-World Environment

Undertaking an engineering survey of existing, real-world asset e.g., of a building or another structure, maybe a time consuming and often cumbersome process. The goal of such a survey is to capture information about the physical characteristics of a given object or structure [1, 155]. This extracted data can be later on used to plan necessary maintenance and conservation works and to estimate the costs of such endeavours.

In some cases, such data cannot be retrieved directly from the construction documentation and design plans as these may be incomplete, contain errors, or not be even available. This is often the case when dealing with heritage sites where an engineer may be tasked with measuring the height or width of a given building or other structure or dimensions of their individual elements (e.g. racks, towers, rooftop surface). Frequently, the ancient heritage monuments and landmarks, such us hoary places of worship (Fig. 4.1), does not have such data available on the spot or this data has not been documented thus far. Similar needs may arise when we are dealing with constructions or structures that occupy large areas, contain hazardous zones (Fig. 4.9) or are in remote locations.

In all of these cases, it may be more feasible and safer to first prepare a digital twin [35] of a given asset. Such digital representation, if of high fidelity, can be used to greatly simplify the task of conducting an engineering survey.

In the situation described in here, an engineer is tasked with estimating the costs of the rooftop replacements of a heritage building, i.e. the Archcathedral Basilica of St. Kostka in Lodz, Poland. As it can be seen in Fig. 4.1, the Archcathedral posses large, rather complex rooftop structure with lots of smaller features and details typical for heritage buildings from its era. Due to the age of the building itself i.e. it was completed in 1912, prior to two World Wars, the existing documentation may be incomplete, inaccurate, or no longer be available. In such a case, the normal order of business would be to (i) close areas near the building, (ii) put in place a scaffolding, and (iii) manually measure the required rooftop dimensions and assess its current condition. This process may not only by be lengthy and costly but may also bring additional risks to the workers and pedestrians as well as being heavily depended on the weather conditions.

Instead, this study have used a photo-realistic digital representation of the Archcathedral generated using a photogrammetry technique. As it was prepared using drone-captured imagery data, there was no need to close nearby areas to set up the scaffolding, and the entire process was efficiently conducted in a short amount of time. In the case of the Archcathedral's building it took six independent drone flights with the total flight time estimated about 3 hours. Including the test flights, equipment preparations and flight trajectory planning the data acquisition took less than a day. This 3D model was then plugged into the VR-based environment where it could be safely inspected and measured by domain-experts from the safety of their own offices.

4.5 Observational Study

We carried out a small user study with six domain experts, all of whom had an engineering higher education background and years of experience in acquiring accurate measurements of physical buildings.

Given the importance of domain experts for this particular application, we opted for an observational study focusing on a qualitative approach with some ancillary quantitative data. Such evaluation methodology is recognised, in for example, Lam et al. [81], as one of the commonly adopted user research methods among visualisation researchers.

4.5.1 Study Participants

The study involved six participants, hereinafter referred to as P1–P6. They all reported various, however small or non-existent, levels of experience with both the VR environment and the hand-tracking interaction method. Here, we present a short description of each participant's professional background:

Participant 1 (P1): reported basic prior familiarity with VR and no previous exposure to any hand-tracking technology. She was 38 years old and had for the past six years has been an office worker in an organisation focused on technical inspections for the safe operation of technical equipment in Poland[3]. She was responsible for organisational knowledge-management, preparation of project documentation and managing employee training. She also reported suffering from astigmatism which had a detrimental effect on how she perceived the gaze-tracking cross-hair.

Participant 2 (P2): was 25 years old and reported having never before used either a VR or a hand-tracking interface. She held an engineering degree in power engineering and had been an intern for the past three months with the same technical organisation as P1.

Participant 3 (P3): reported basic prior familiarity with VR and no previous experience with hand-tracking technology. He was 32 years old and held an engineering degree in power engineering.

Participant 4 (P4): reported previous basic experience with both VR and hand-tracking technologies. He held an engineering degree in mechanical engineering and was 44 years old. He reported having 18 years of experience with CAD systems.

Participant 5 (P5): was 47 years old and reported basic prior familiarity with both hand-tracking and VR technologies. He held an engineering degree in power machinery and had 21 years of related work experience.

Participant 6 (P6): reported neither using VR nor hand-tracking technology. He was 57 years old and held an engineering and a master degree in civil engineering. For the previous 10 years, he had been working as project manager on construction sites. He reported sometimes perceiving a double cross-hair in the head-set and usually wore corrective glasses.

4.5.2 Study Design

Training Video Clip

After signing the consent form, participants were shown a training video recorded from a first-person perspective representing the user's field-of-view. During the video, participants could ask questions if something was unclear.

[3] Urzad Dozoru Technicznego *eng. Office of Technical Inspection* (UDT) is an EU Notified Body No. 1433 (www.udt.gov.pl)

Questionnaires

To assess the level of potential symptoms of the simulation sickness that the VR environment might cause, each of the six participants were asked to fill out a Simulation Sickness Questionnaire (SSQ) [74] before and after using the VR headset. In addition, the participants were also asked to fill out the NASA Task Load Index (NASA TLX) [53], the Flow Short Scale (FSS) [112] and the Ipgroup Presence Questionnaire (IPQ) [61] questionnaires immediately after they had completed their assigned tasks. The NASA TLX questionnaire was used to estimate the level of perceived cognitive while carrying out a task within our system. The purpose of the FSS and IPQ questionnaires was to assess the VR environment's impact on perceived flow [29] and presence respectively.

Tasks

The participants were asked to carry out two tasks in the VR environment while standing and wearing the VR headset. The instructions were: (Task 1) *Please take measurements of the appropriate dimensions of the roof using the interaction methods presented so that you can calculate the approximate rooftop surface of a given object.* (Task 2) *Please take all other measurements that you consider relevant concerning the surveying of the asset.*

The purpose of Task 1 was to allow the participants to focus on a task while learning the interaction techniques. Task 2 allowed us to observe how they would use these newly acquired skills for surveying without biasing them. As such, the measurements taken in Task 1 were not meant to be detailed, especially since the Archcathedral rooftop (Fig. 4.1) is a very complex structure. Rather, the aim was to give the participants an idea of how these interaction techniques can be used in their everyday work.

The tasks were elicited from discussions with engineering practitioners, before being given to the domain-expert volunteers. This was done to ensure the given tasks are representative tasks that experts carry out in their day-to-day work. In addition, the second task provided the expert-users freedom to measure whatever they were interested in measuring in the context of conducting engineering surveys of a given asset. Our system is using a photo-realistic, georeferenced, and calibrated 3D model (Fig. 4.1), therefore, the task of measuring any part of the model supports the users in capturing information about a real-life structure.

As the design space is large, we decided on a task that would be meaningful for the experiment participants and yet would allow us to trim the design space by observing users' behaviour. The task would also have to be simple enough to not force the users to spend a substantial amount of time training, as their increased fatigue would have impacted the experiment's results. Furthermore, the second task was flexible enough to allow participants to decide for themselves what they are interested in measuring when immersed in the virtual environment. This approach allowed us to capture more insights into how the users would like to use such systems.

Experimental Phase

The participants were standing or walking slowly within a limited space constrained by the length of the HMD cable. Whilst carrying out the task the participants were asked to think aloud and constantly communicate with the researcher about what and why they are trying to do at the moment. They were also encouraged to ask questions if they were stuck or if they did not know how to proceed further. The researcher was always present when a participant was wearing the HMD and could observe both the participants' behaviour and what they were experiencing in the VR environment by observing the participant's field of view simultaneously streamed and mirrored to a nearby computer display. The main part of the study was complete when the participant took all the required measurements or the researcher or the participant concluded that their understanding of the interaction techniques (Fig. 4.4–4.7) was sufficiently high for the more detailed rooftop measurements (Fig. 4.7) to be easily carried out without assistance from the researcher.

Follow-Up Discussion

The experimental part of the study was followed up with a semi-structured discussion between the participant and the researchers.

Language Translations

The study was carried out in Polish. The questionnaires were translated by the researchers who sought external verification of the translation from a native-speaking researcher with expertise in research in psychology and human-computer interaction.

4.6 Results

4.6.1 Questionnaire Survey Results

The number of participants was small. Hence, a statistical analysis is inappropriate. Therefore, the descriptive quantitative statistics below only refer to the sample of participants for this

Table 4.1 The participants' results are presented next to the total possible score for each questionnaire. The second column reports the approximate time spent in VR by each participant. For SSQ [74], we only report scores after the experiment was completed. Here, half of the participants (P2, P3, and P5) reported oculo-motor strain levels of 4.0 before the experimental phase commenced. The numbers in parentheses in the NASA-TLX [53] column are repeated decimals. In case of the IPQ [61] and FSS [112] the higher the score means higher experience "feel of presence" and higher engagement respectively, as opposed to NASA-TLX and SSQ where we want to observe the lowest cognitive workload and sickness symptoms.

Questionnaire Survey Results							
Participant	Duration	IPQ	SSQ		FFS		NASA TLX
no	*minutes*	*average*	*nausea*	*oculo-motor*	*flow*	*anxiety*	*total score*
P1	25 : 38	3.50/7	0/27	4/21	6.2/7	4.5/7	59.(3)/100
P2	34 : 08	3.71/7	0/27	2/21	6.5/7	5.5/7	30.(0)/100
P3	17 : 53	3.14/7	0/27	4/21	4.2/7	5.0/7	48.(6)/100
P4	16 : 38	3.64/7	0/27	2/21	5.2/7	2.0/7	13.(0)/100
P5	19 : 80	4.00/7	0/27	1/21	6.2/7	3.5/7	42.(0)/100
P6	22 : 37	2.79/7	0/27	1/21	4.1/7	2.5/7	31.(6)/100

study—the domain experts, and we do not attempt to generalize the findings to a wider population.

Notably, SSQ [74] results revealed only very slightly elevated levels of oculo-motor strain among all participants, whereas we observed no effects of nausea. FSS [112] results revealed the lowest observed flow score was 58.57% (P6) with 92.8% (P2) being the highest which suggests a high level of engagement. An analysis of the NASA TLX [53] results revealed that P1 reported a 59/100 score whereas all the other participants (P2–P6) reported scores lower than 50/100 with P4 reporting the lowest score of 13/100. These results suggest relatively low levels of cognitive load across the participants. For the IPQ [61] analysis, as in Schwind et al. [125], we averaged the seven-point scale by the number of questions answered by the individual participants. The IPQ results revealed the majority of the scores approaching 50%, with the lowest observed 2.79/7.0 (P6) and the highest 4.0/7.0 (P5). This demonstrates a passable level for participants' perceived sensation of presence. All of the results from the questionnaires surveys can be found in Tab. 4.1.

While the participants were in the experimental phase we gathered additional quantitative feedback (Tab. 4.2). This data has to be analyzed with caution as it was not acquired in a timed, fully controlled experimental setting, and was gathered from a small number of domain experts. It was gathered to gain insights about to what extent the participants used the available techniques to manipulate objects (Fig. 4.5). As such, we were interested in the total angle by which they rotated the object, as a small rotation may be mistakenly

Table 4.2 Quantitative feedback: (r) the model's maximal rotation angle in degrees and (s) minimal and maximal building scale. The **[distance]** column shows the total displacement of the model by each participant.

Participant		min	max	distance
P1	(r)	-	99.4°	21.3
	(s)	.698	38.26	
P2	(r)	-	66.7°	66.5
	(s)	.618	347.8	
P3	(r)	-	71.6°	21.1
	(s)	.282	4.475	
P4	(r)	-	66.0°	6.24
	(s)	.779	4.877	
P5	(r)	-	129°	32.9
	(s)	.090	1.359	
P6	(r)	-	109°	11.6
	(s)	.310	1.777	

applied while moving the model with both hands. We were also looking for the maximal and minimal scale of the model applied by the participants. In addition, we investigated the total displacement distance of the model, that is, the absolute Euclidean distance of the model's movement trajectory. This data indicated that P2's behaviour slightly deviated from the other participants. Comparing to the other participants, she had, at some point, substantially enlarge the model zooming in on it (Tab. 4.2). She also moved the model around over twice as much total distance as the next participant (Tab. 4.2). A possible factor may be her relatively young age and fewer years of professional experience.

4.6.2 Observed Behaviour

The tool was designed for a specific purpose, that is, to conduct inspection and surveying of an engineering asset. Thus, we focused on leveraging the domain-experts and engineering practitioners unique expertise and views as much as possible.As it is difficult to recruit volunteers with enough expertise and desired background, we were determined to obtain as much information as possible within the time-frame devoted to the study. Therefore, we decided to use a mixed-methods approach to gain as deep as possible an understanding of how the domain-experts would behave when using immersive interfaces to conduct an engineering survey of a 3D photogrammetric model. This approach included a think-aloud protocol as well as a semi-structured interview carried out immediately after the second task was completed. This approach resulted allowed us to analyse and report on a number of our own observations, as well as on comments made by the study participants, to reflect on the system's usefulness and the users' needs and wants with respect to adopting such tools in their own day-to-day work. The captured quantitative data allow us to confirm users' comments by inspecting logged data related to their execution of the tasks. For example, we could inspect by how much a participant had shrunk or increased the size of the model in order to place the ruler's marker.

We analysed the video recordings that captured both the participants' gestures and the computer display streaming the participant's field of view. The video footage of P1 excluded nine minutes in the middle, which was lost due to a battery outage. We analysed the videos to first identify a number of modalities along which we were focusing when re-examining the recorded data for the second time. As such, we were particularly interested in studying the following patterns:

Users' Movement Patterns

We were interested in how the user move (such as moving vs. standing in place) in the real-world which was instantaneously mapped onto their position in the VR environment. Half of the participants (P1, P2, P5) walked around, whereas the others (P3, P4, P6) preferred to remain in a fixed location while using the system. It might be related to the fact that these participants (P1, P2, P5) chose to operate on a largely magnified model in VR space (Tab. 4.2), which in turn also made them stretch their arms and hands to place the markers more frequently than the other participants.

Model's Movement Patterns

After the initial moments of the experimental phase when the participants were told to familiarise themselves with the interaction techniques, we observed that all the participants have frequently used the available manipulation methods, such as (r) rotation, (d) movement and (s) scaling (Tab. 4.2). With the exception of a single participant (P2), the participants typically chose to manipulate the model by rotating it by small angles, moving it by small distances, or changing its size by a relatively small amount. The participants' manipulation outputs were probably constrained by the length (for movement and rotation) and degrees of freedom of a human arm (for movement). We also observed, that P2 who, for a while, extensively increased the model's size as compared to the others (Tab. 4.2) seemed to generally prefer to work on a larger model. However, when she (P2) too promptly increase the magnitude of the model, it caused her to startle, which most likely speaks to how realistic the model appeared to her. She (P2) also commented that *[it's] intuitive, its a matter of familiarity with it, I think, because I had a problem with how to rotate it whilst it change* *its size simultaneously*. No other participant has had troubles with it. For example, P6 commented that *I liked rotating, moving, zooming in and out*.

P3 on the other hand, only changed the model's size once when he selected an appropriate magnitude (for him). He also wished to rotate the model about more axes and not just in the $X-Z$ plane. A similar question about the rotation limits was raised by P5 as well: *Can it be rotated in any plane?*

P5 also tried to increase the size with spreading his right-hand fingers similarly to the gesture known from touchscreen interfaces. He suggested that in such case the user*[...] could enlarge with one hand and, for example, reduce* [the model's size] *with the other* [hand].

Measurement Patterns

The participants (P1, P2, P3, P4, P5) chose to connect the rules and take the measurements only after they felt that all the necessary markers were placed. Only in a single case (P6) a participant decided the take the measurements in two smaller chunks. However, the measurement patterns were still similar to the other participants.

Some users (P1, P2) preferred to look for the previously generated and not-used markers instead of generating new ones. After being suggested that it may be easier to generate a new marker they chose to do just that the next time. Moreover, P2 chose to generate a number of markers first to place them one by one later on, whilst the other participants preferred to generate no more than a few markers at once.

Gesture Usage

In the beginning the participants had to be frequently reminded about the *thumbs-up* gesture (Fig. 4.4(d)) used to release the handle over a selected ruler's marker. P5 suggested that when one hand is placing the marker the other hand can be used to make the *thumbs-up* gesture.

All participants kept pointing with their fingers or entire arms towards the model to explain what they did or what they wanted to do to further illustrate what they were communicating. This behaviour confirms the high level of engagement indicated in the questionnaires when immersed in the system.

In a single case (P3) a participant kept his left hand alongside his body during the majority of the experiment. P3 also suggested, that he would like to see *The Projections of the point's coordinates on the coordinate system's axes*.

We observed that when the participants placed the marker on the Archcathedral Tower (Fig. 4.1), all participants extended their arm to place the marker on top of it. This happened despite all participants being familiar with the manipulation methods and knowing that they could easily decrease the size of the model, lower it, to prevent the need to extend their arm.

Gaze-Tracking Usage

As we were using head-tracking as an approximation for the gaze-tracking, there were some issues related to this approach as remarked by the participants. For instance, P1 remarked that: *The worst were the two* [cross-hairs] *that I see all the time [...] they annoy me because I feel confused.* However, she also commented that *[...] I have been diagnosed with astigmatism, maybe that's why? I never did anything with it* [correcting the vision]. P2 remarked when grabbing the markers that *I forget to aim with this circle* [the cross-hair]. P6 commented that he did not intuitively recognised the connection between the glowing effect and the ability to manipulate the glowing marker. He said *At first glance, I did not see this relationship.*

Pop-up Menu Usage

At the beginning of the experiment, P1 kept the pop-up menu continuously visible (Fig. 4.4). When involuntary or voluntary change in the hand gesture occurred she closed and opened her fist to recall the menu. During the experimental phase, P1 frequently chose to keep the pop-up menu visible while simultaneously making gestures with the other hand, such as removing or placing the rulers' markers around the model. Such behaviour was not observed among the other participants with exception of P6 who for a prolong amount of time kept his left-hand in the upper hand position.

The participants frequently forgot that to connect the markers the user has to be in default mode, that is, no buttons should be selected on the left-hand menu.

Further, P5 suggested a vertical heads-up menu instead of the pop-up menu: *Can't such a menu be pinned permanently at the side-corner of the view so it would be visible [...]* [constantly].

Desired Precision of Measurements and Movements

We observed that the main issue with the precision of movements and measurements had to do with the grid overlying the 3D model and a too large radius of the snapping distance. However, despite these inconveniences the participants were able to carry out the necessary measurements. For instance, P1 remarked that *[...] the precision of the movements, I can see that with each such procedure it is getting better and better. Therefore, it seems to me that [...] it is a matter of training. The more a person interacts with this program, just like with a computer game,* [that person] *becomes more efficient and proficient.*, but she has also commented on the other occasion that *[...] the manipulation is not precise enough for me.* P3 commented that *The general idea is good only with precision [...] If it was more accurate, it would be better to move these points* [rulers' markers] *to the places I want, it's a very nice thing for sure.*

4.7 Task Analysis

Due to the limited availability of qualified participants, we decided to use *back-of-the-envelope action analysis* [88] which is a technique used to inform the interface designers about the complexity and sequence of actions that have to be taken by the user in order to complete a given task [88]. Such analysis can help in spotting too complicated action sequences without the necessity of conduction user studies.

The system requirements identified two key, high-level design requirements (**DR1** and **DR2**) that can be also understood as two task that the system has to support. These two tasks, to some extent, blend into each other or are usually executed in intertwining steps. For instance, to take the measurements (**DR2**) it may be necessary for the user to manipulate the model (**DR1**) in order to decide which measurements have to be taken. These two tasks (**DR1–DR2**) may have to be executed multiple times. As such, our action analysis [88] was focused on both these design requirements. When conducting the task analysis, we have identify potential improvements of our interface.

First concerns the visual localisation of the object that user want to manipulate. In our case, the 3D object is initially placed in front of the user's field of view. However, once the model was displaced to a position in the 3D space outside of the user's view, the system does not help in the localisation of the object. Here, we could potentially include a feature that informs the user in which direction from a current position of the user's gaze the object was translated. We could possibly add a re-call button to the left hand pop-up menu that would move the model to a position in front of the user.

Second concerns the pressing of a button on the menu to select the object's manipulation mode (see Fig. 4.4 (a)). As observed in the user study, the object's manipulation is one of the most frequently used features. Hence, we could additionally set it to be the automatic mode when the user does not have any options selected (i.e. any buttons pressed) or if the user is idle for a fixed amount of time.

Finally, while analysing steps needed to take a distance measurement we can notice that some of them are repeating. Even though such setup gives the user more flexibility it perhaps may be more feasible to provide the user with an option to generate already connected pairs of markers. Then, such markers can be moved independently while the system automatically recalculates the distance measurements by repositioning already connected markers. Such an option would simplify the task of placing the initial measurement wire-frame.

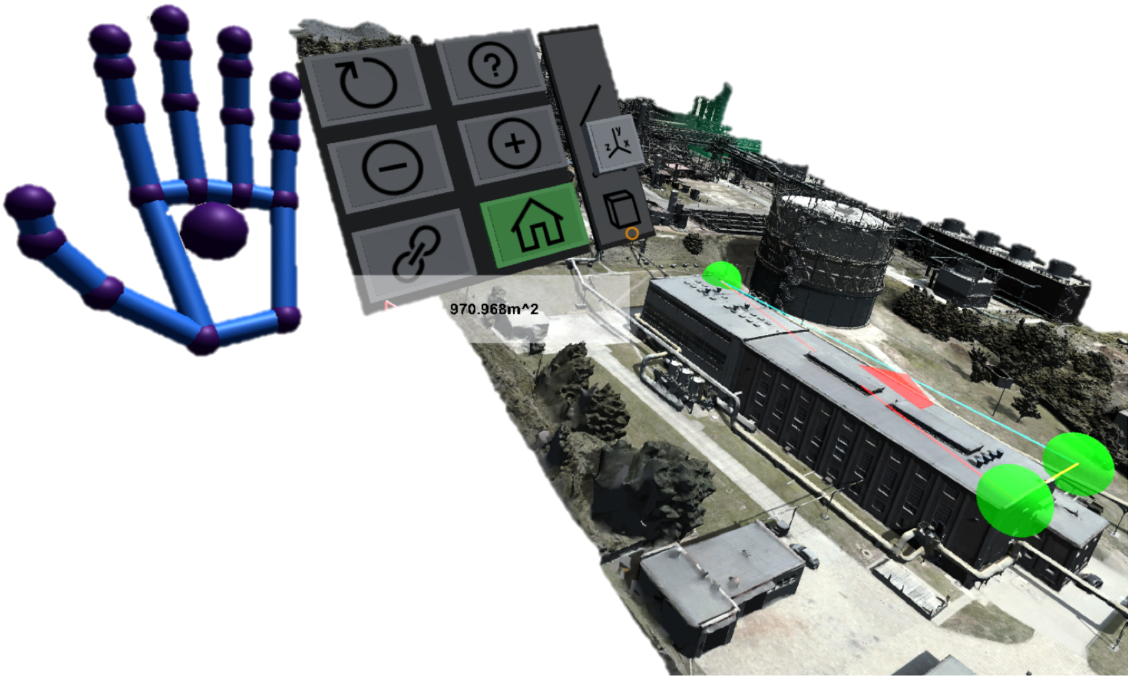


Fig. 4.9 The surface measurement of a building rooftop at the plant. The surface is approximated by the surface of a triangle whose vertices are given by the three rulers' markers (spheres) placed by the user in the rooftop corners.

4.8 Surface and Volume Measurements

The natural extension of the dimension measurement capabilities that were tested during the observational study are the abilities to capture surface and volume measurements as well. To carry out a surface or volume measurement, the user has to first use the additional slider menu (Fig. 4.9 or Fig. 4.10) attached to the left-hand pop-up menu to select the desired measurement type: (i) surface; or (ii) volume. Once selected, the user can now place the ruler's markers around the model. A minimum of three (for surface) or four (for volume) markers have to be placed and connected to carry out the measurement.

In both cases, the measurements themselves will be taken and presented on the HUD in chunks (Fig. 4.9 or Fig. 4.10). The surface measurements will be reported in a form of triangulation of a surface, whereas the volume data will be captured as a series of volumes of tetrahedral. This allows the user greater flexibility to measure objects of higher complexity that are also constructed from triangular meshes.

4.9 Discussion

Thanks to the immersiveness bestowed by VR, there is a substantial difference between how the user perceives, operates, and behaves within an immersive environment as opposed to a traditional workstation setup. There is a number of factors that contribute to this differentiation. For instance, various workstation-based specialised engineering software (such as CAD packages) have been in constant development for decades, which has resulted in well-established and mature technology. Engineers are also regularly taught how to use these programs and they are in continuous use throughout the users' professional careers. Hence, this prior experience in established practice has most likely biased domain-expert users towards the familiar 2D environments where they are able to interact using familiar interaction techniques facilitated by a computer mouse and a keyboard. Due to this bias, the fact that keyboard and mouse are not typically used in VR and the reality that VR technology still not fully mature, it is not, at this current stage of VR technology, meaningful to directly compare and contrast these two interfaces.

The main goal of the study was to consider the feasibility of using a gesture-controlled environment coupled with photo-realistic 3D digital twins of real-life structures to conduct a relevant task that domain-experts would be interested in when working with such software. 3D photogrammetric models allow a very close and detailed representation of real-life objects, and especially has an ability to capture physical information about them, such as their dimensions or their surface structure drawn from imagery information, including, for instance, cracks and discolouration of a rooftop. Such information can be used by experts to reason about the physical state of a given structure. This is not possible when working with computer-aided design (CAD) models, which we have investigated in our earlier studies [154]. Even carefully crafted CAD models do not represent the current visually-visible state of real-life structures as they are not generated based on recent captured data. Hence, they cannot be used to conduct an engineering inspection of an existing asset. In contrast, 3D photogrammetric models can be useful in effectively extracting such information from models instead of conducting a costly and time-consuming outdoor inspection. Hence, the goal of this study was to explore and trim the vast design space of designing immersive interfaces with such a purpose in mind by reporting on how domain-experts would behave when using VR environments to take measurements of photogrammetric models.

Observations of the participants' performance and behaviour, as well as their qualitative feedback, highlight the strengths of the bimanual manipulation of the 3D photogrammetric model. These manipulation methods include: (1) move and reposition the model in the 3D space; (2) decrease or increase the model's size; and (3) rotate the model in the X-Z plane. The interaction approach afforded by the bimanual manipulation of the objects allowed for natural and intuitive techniques. The users quickly became fluent in these actions and, after a very short period, did not require additional guidance from the researchers in order to effectively execute these techniques.

Some of the features, such as the snapping-grid, were not favoured by the participants. It seems that this effect was magnified by their professional and educational background in engineering where the ability to take very detail measurements is crucial. (P3: *In general, the idea is good, only the* [issues] *with precision* [of measurements].) However, as some of them commented, the snapping grid can be useful in other ways. For instance, P1 had an idea of using the grid points as control points of a checklist used by an installation inspector while executing an actual task. The snapping grid could also be useful to fix the marker's position so that it would not follow the user's hand when the hand is moving to another location and the user's intention is not to move the marker along with the hand movement. This would eliminate the need for using the *thumbs-up* gesture. Such a grid could be recalled and hidden on the user's demand, hence allow more flexibility with marker placement when needed. This suggests that the snapping grid idea, if tweaked to take into account the domain experts' feedback, can serve as a useful tool. A snapping grid may also positively impact effectiveness and ease of use when measuring surfaces and volumes. Here, the user has to place and connect a substantially larger number of markers. In such cases, it is viably feasible for the system to provide the user within an initial estimation of where the next points should be placed, as well as automatic connection suggestions with nearby points generated as a group rather then individually. We conjecture that this feature design would increase performance.

The observed behaviour, and questions asked during the tasks' execution, suggest that the system needs a clear distinction for the measurement mode. It could be applied, for example, by showing an additional button on the menu, with the default mode set to the model's manipulation mode (which enables the gestures shown in Fig. 4.4). Further, P5 commented that he would prefer to have a view-wide vertical menu on the top-to-bottom HUD. However, no other participant made a similar suggestion.

Moreover, even though each measurement is automatically saved to a file and the three latest changes are visible on the HUD, P4 wanted to see more of them at once and have some constant fixed measurement as a reference frame. P3 suggested that the markers have their

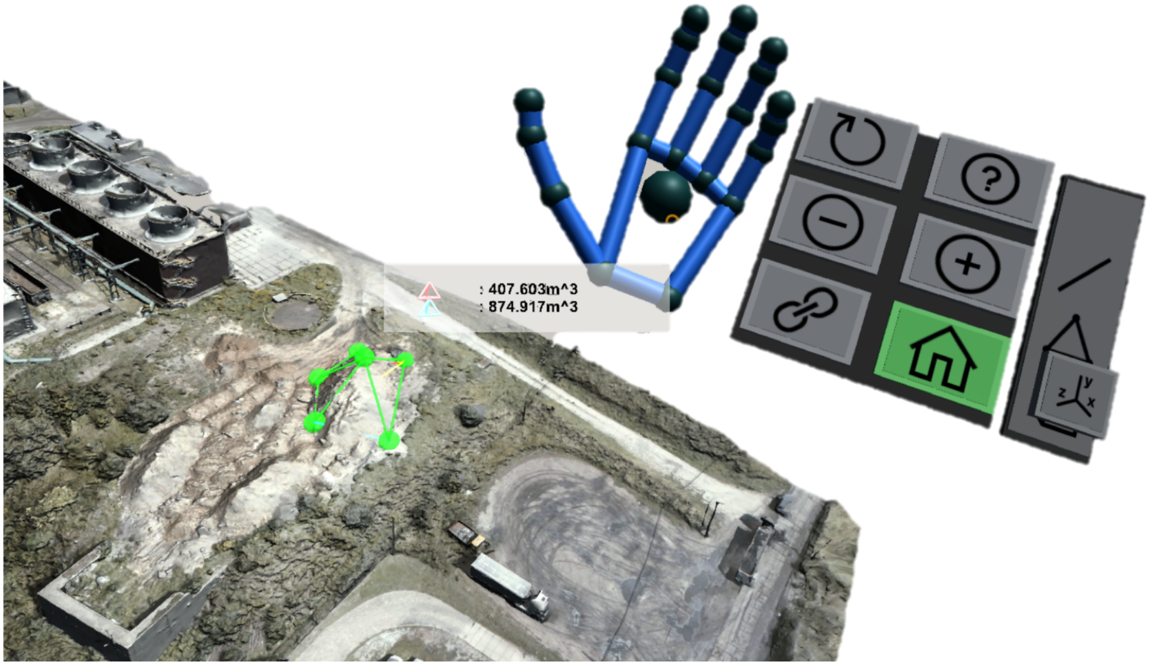


Fig. 4.10 The volume measurement of a pile of sand taken over a photogrammetric model of the plant. The user placed several ruler markers (spheres) and obtained volumetric measurements of two convex polyhedrons (tetrahedrons).

coordinates projected as vectors on the model's local coordinate system. To some extent, these suggestions can be fulfilled by allowing the user to recall a wire-frame of the rulers connecting a set of essential mesh vertices on-demand. These, in turn, can be determined, for example, by using a convex hull algorithm or any other form of more advanced mesh analysis using computer vision.

From our observations and with comments made by P1 and P2, it appears feasible to explore a bimanual manipulation technique in which each of the gestures have a different meaning depending on the hand used. It allows users to rotate the model with one hand while simultaneously move the model with the other (P1); or use one hand to decrease and the other to increase model scale (P2).

4.10 Design Implications

Here, we distill design implications for a system for conducting engineering surveys of a real-life structure or object in VR.

DS1—Model manipulation: The system should assist the user in effective manipulation of the 3D model by allowing the following three main operations on the model: (1) move and reposition the model in 3D space; (2) decrease or increase the model's size; and (3) rotate the model by any chosen virtual axis. Bimanual methods of executing these three manipulations seemed to be easily understood and accepted among the study participants and at the same time led to satisfactory results.

DS2—Model measurement: The system should support the user in taking three basic (direct or indirect) measurements types: (**DS2D**) distance or length of any of the parts of the model; (**DS2S**) surface area (2); and (3) (**DS2V**) volume. Moreover, as to some extent these measurements can be thought of higher-order extensions of each other, that is, **DS2D→DS2S→DS2V**, the interface should allow an efficient and easy to learn method for switching between these measurements. When starting from the measurement requiring fewer points (distance) and switching to more complex measures (surface and volume), each measure has only a small additional cost (more markers), or no extra cost whatsoever when going the other way as these markers would already be placed in the 3D space. Here, the engineers may be interested not only in the overall surface area or volume but also in details of the measurements on which these calculations are based upon, such as the lengths of a triangle's or a pyramid's edges. In the VR environment these can be easily achieved by extending the left-hand pop-up menu with a slider (Fig. 4.9 and Fig. 4.10) that provides an easy to learn and efficient method for switching between the measurement types without compromising previously taken results. Additionally, the model structure itself, that is, the triangular mesh, can be leveraged further by providing automatic suggestions on placing the markers in pre-selected spots around the model.

4.11 Conclusion

The experiment discussed in this Chapter have studied bimanual interaction techniques for supporting engineers measuring distances, surfaces and volumes of photogrammetric 3D structures using a VR environment. The efficiency and easy of use of the techniques were studied in a case study with six domain experts.

Based on the observations of user behaviour and the participants' own comments, it can be concluded that enabling domain experts in surveying to use their own hands to directly manipulate—grabbing, moving, rotating and scaling—a 3D model in VR is perceived very favourably by such users. All of the participants were promptly able to use their own hands to directly manipulate and take engineering measurements of a complex real-world model in the VR environment. The domain experts also noted that their performance would likely improve with practice. However, the design of the left-hand menu and the overall measurement toolkit should be refined and extended to fulfil the domain experts' needs and wants.

Furthermore, as remarked in Chapter 2, even though there is prior work concerning VR environment populated with photogrammetric models, [44, 90, 40], only a handful of articles is specifically concerning interaction in such context. Here, this work focuses on how bimanual gestural input can be leveraged to obtain an effective system for conducting engineering surveys of a real-world asset. The system described in here uses only single sensor, a Leap Motion [84], attached to the front of the HMD. It provides an alternative set of single and two-handed gestures supported by the Leap Motion Software Development Kit [84]. Using the Leap Motion [84] controller allows building systems that can recognise more complex gestures. For example, Clark et al. [22] developed a real-time hand-gesture recognition system for VR using machine learning methods. However, as shown in the qualitative study, the basic set of gestures supported by the Leap Motion sensor is sufficient for a plausible and effective interaction in VR.

The case study explored interaction techniques that allowed a domain expert to capture not only dimensions but also, as a by-product, calculate the surface and volume measurements of the desired model or its individual parts. Subsequently, after analysis of the study results, the system's toolbox was extended with new functionality that allows the users to take surface and volume measurements (Fig. 4.9 or Fig. 4.10) without the need of calculating these values themselves. The way in which these features were implemented flows naturally from the fact that the models are constructed from triangular meshes, and from the initial method of placing and connecting two points in a 3D space with a vector used to capture the distance measurements. Hence, the user does not have to spend time learning intricate novel interaction techniques and there is no need for the system to support elaborate gesture recognition of complex gestures.

The use of photogrammetric models that recreate, in fine detail, real-life objects has great potential as a means of obtaining high-quality VR content. In addition, such models can help bring to life objects and structures well-known to the users from their everyday life, thus enhancing the experience of the VR headset user. This case study with domain experts indicate that it is viable to support engineering measurements of such complex 3D structures using easy-to-use and efficient interaction techniques. In turn, these observations and results further support the suggestion that photogrammetry has the potential to become a very feasible option for generating digital twins of existing, real-life objects, structures or even larger areas.

Chapter 5

Virtual Reality Analytics in Computational Design

Computational design methodologies and special optimisation tools are common practice in modern engineering. The primary benefit of such approaches is the automated exploration of complex design spaces, which was previously not possible. However, this advantage creates user interface challenges on how to support manipulation and analysis of high volume and high complexity multidimensional datasets. While even formulating a real-world engineering optimisation problem is a difficult task, the inherent additional difficulties in extracting understanding, causality, and rationale of discovered information is exacerbating the demands on the user.

The ultimate goal in this work is to assist users in understanding and then communicating the complexities of the engineering design task in question and thereby supporting the decision-making process. As a step towards this goal this work presents the system design of immersive parallel coordinates plots (IPCP). Prior work [75] has demonstrated that visualising scientific information in an interactive and engaging environment is beneficial to users' decision-making abilities. We hypothesise that further benefits of visualisation can be realised using emerging 3D immersive environments, such as Virtual Reality (VR) and Augmented Reality (AR).

This Chapter explores how VR can be used to support visualisation and analysis of computational engineering design datasets. In addition, we discuss the positive and negative aspects of the application of VR in the decision-making process. The overall task we study is the identification of clusters and sets of solutions in an optimisation study for the aerodynamic design of compressor blades [76, 77]. However, we do not consider the physical interpretation and understanding of the designed compressor blades. We rather consider the physical nature of multi-objective optimisation data. In such datasets, the input parameters express the design

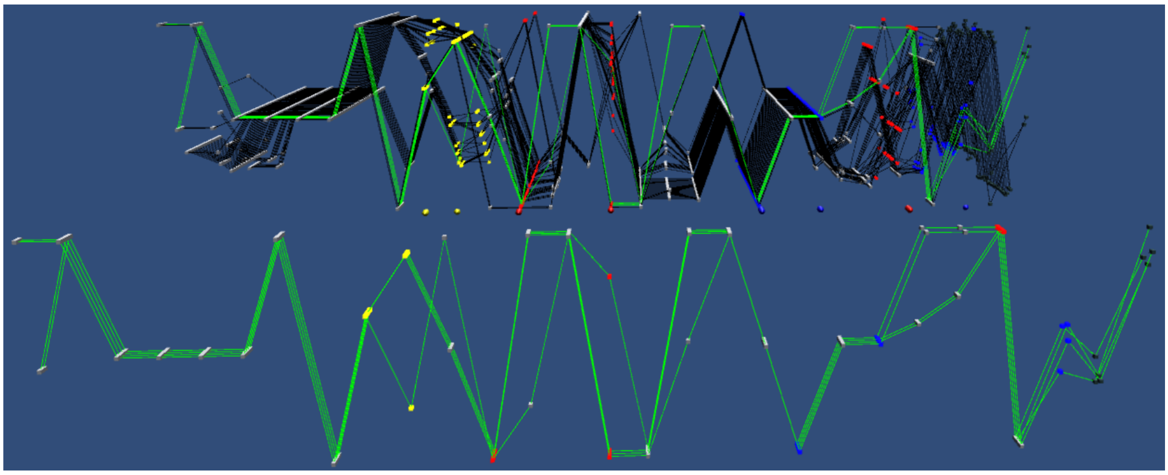


Fig. 5.1 The user's field of view in the VR environment. Selected data items are copied to a separate view (in green and in front). The copied subsection of data items, the main IPCP and the 3D scatter plots supports the *linking & brushing* principle i.e. the manipulation of one of the elements is constantly reflected on the other elements of the visualisation as well.

parameters of the design optimisation study, and the output parameters express the objective functions and constraints.

This work does not address the classic challenges in using parallel coordinates plots (PCP) for visualisation and analysis of multidimensional scientific data. Instead it explores how to build a system that can support the design process and investigate the usability of emerging immersive technologies for visualisation of, and interaction with, scientific information. This allows to assess whether the same challenges remain, or new ones emerge. Furthermore, the exploration of engineering design computational data is frequently centred on 3D visualisation of the designed product. As such, an immersive interface offer a highly promising method for analysis and exploration of such data alongside with simultaneous visualisations of individual product designs. Part of the motivation of this work is to expose and exploit such strengths of VR technology. To this end, this Chapter presents the design of our VR-based IPCP system that is supported by a series of formative studies. A description of the first thereof which briefly discusses the observational user study, results of which were used to trim the vast design space, can be found in Tadeja et al. [150]. The underpinning VR system that included the basis of interaction techniques i.e. the gaze-tracking and racy tracing facilitated by a cross-hair was discussed in Tadeja et al. [156, 157] and is built based on the *Unity VR Samples Pack* [166] that utilises the well-know concept of a cross-hair to simulate the user's gaze (for instance, see orange circle in Fig. 5.4) and *Oculus Utilities for Unity* [106] together with some elements from the Unity asset store packages [47].

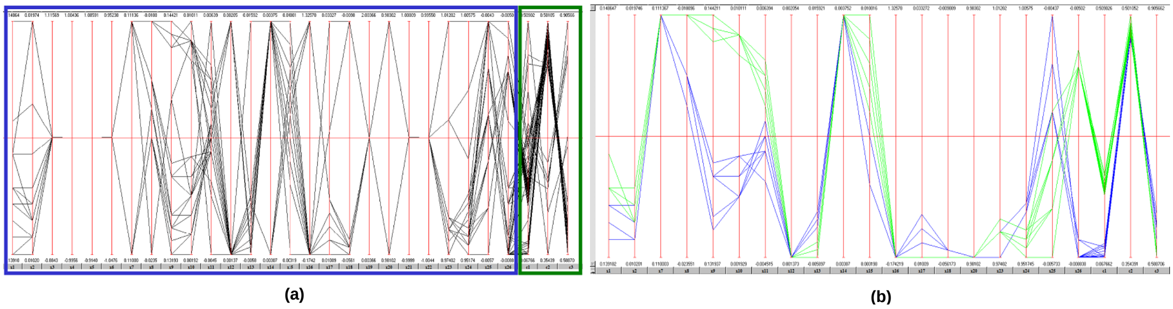


Fig. 5.2 (a) the standard 2D PCP visualisation of the dataset. The blue frame covers the *design parameters* (first 26 dimensions) and green frame covers the *objective functions* (last 3 dimensions). (b) the two patterns (in blue and green) originally identified by domain experts. Reproduced from Kipouros et al. [75].

5.1 Background and Problem Context

The sample dataset originates from Kipouros et al. [75] and is the result of a design optimisation study for the 3D geometrical design of compressor blades subject to many equality and inequality constraints. In this study, we use the data produced for a three-objective function formulation of the optimisation problem. Further details can be found in Kipouros et al. [76] and Kipouros et al. [77]. This dataset describes the Pareto front, which contains a set of 54 equally optimum design configurations, and the 29-dimensional data items categorised as: *design parameters* (26 dimensions) and *objective functions* (3 dimensions). The 2D view of the data is presented in Fig. 5.2 (a) whereas Fig. 5.2 (b) shows the two patterns of interest A (in blue) and B (in green) composed of 11 and 19 data items respectively that were identified in this dataset by the domain experts. The user's view through a VR headset in 3D is presented in Fig. 5.1.

5.2 System Design

5.2.1 Overview

The overall objective is to design a visual analytics system for IPCP for engineering design processes, supporting efficient identification and comparisons of patterns of interest in a multidimensional dataset. This is a high-dimensional design space with many controllable and uncontrollable parameters. Controllable parameters include rendering parameters, such as colours and shapes, visual/auditory feedback design, interaction techniques, and choice of headset and hardware controllers. Uncontrollable parameters include properties about the datasets, such as size, density, and correlation structure, and the degree of domain knowledge and expertise of the user.

As a result of the many controllable and uncontrollable parameters, we have adopted an iterative process of first building an early system that provided direct transplantation of PCP to VR. This early design was then used to carry out qualitative user studies (partially described in a previous poster publication Tadeja et al. [150]), briefly outlined in Subsection 5.2.2. We use the information distilled from the user study along with knowledge of the tasks, expert advice, and technical constraints to arrive at a set of design principles, which guide our task analysis in Subsection 5.2.7. In the task analysis we break down each task into functions and translate each function into a solution based on technical constraints, expert advice and information gathered from the qualitative user study [150].

5.2.2 Qualitative Study

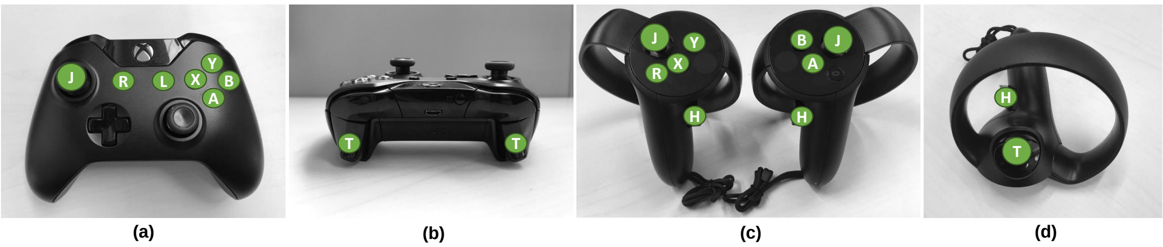


Fig. 5.3 Various VR controllers : (a–b) an Xbox gamepad controller; (c–d) Oculus Touch. The green markers with accompanying letters are showing mapping between the action-buttons on both types of controllers. The description of user's movement in the 3D space together with interaction details can be found in Tadeja et al. [156, 157].

To help inform the design we carried out a qualitative study with seven participants (hereafter referred to as P1-P7). More details about this study can be found in Tadeja et al. [150]. The VR-based IPCP visualisation used in the study can be seen in Fig. 5.1. Users could interact with the system using a mixture of gaze-tracking and Xbox gamepad controller, as shown in Fig. 5.3 (a–b). The colour-coded cones[1] were used to denote both the selected dimensions over the IPCP and axes directions at the scatter plots [156, 157], see Fig. 5.3.

Although the visualised dataset comes directly from the specific domain–the aerodynamic design of turbomachinery components–we wanted to pinpoint the general underlying design principles that can be used in future more advanced versions of the IPCP. Therefore, we broadened our sample of participants to include both fields of expertise and prior familiarity

[1] W. Kresse, used under CC BY-SA 3.0; http://wiki.unity3d.com/

with PCP. The volunteers self-assessed their level of expertise with regards to VR and PCP using a Likert-like scale. As the colour-coding has a crucial role in the visualisation, participants were pre-screened with Ishihara's tests [65], which indicated no colour deficiency.

On average, participants needed 25 minutes during the training phase and 15 minutes for the study itself. To offer insight into any adverse physiological effects, volunteers were instructed to take the simulation sickness questionnaire (SSQ) [74] before and after each phase (i.e., training or study) to see to what extent and if at all the visualisation impacted negatively on their perception of comfort. Participants were reminded that they should stop the experiment if they experienced any of the symptoms described in the survey. However, none of the participants decided to stop and discontinue the study.

Before the experiment commenced, each participant sat through a demonstration of the IPCP technique, the tool under study and all the possible ways in which the user could interact with the system. Moreover, while listening to a verbal description, participants could track in real-time the instructor's actions and their corresponding results on a computer.

Our participants had varying levels of prior familiarity with the PCP and VR. To level the playing field, we provided each volunteer with training data visualisation. This process took between 18 and 36 min, depending on the individual participant's needs, to achieve familiarity, as judged by the researcher. The tasks included: a) selection of both data items and dimensions on the PCP; b) selection of data points and clusters on the scatter plots; c) visual analysis of the relation between the visualisation elements; and d) rotation and movement of a scatter plot or generation of more than a single such 3D projection at once. Familiarisation with the system was especially important, since the training may be very beneficial for the participant's grasp of the PCP technique, as remarked by Shneiderman et al. [138]. For this purpose, we used a portion of a real-world dataset of 80 multidimensional data points, each consisting of 22 dimensions (20 inputs and 2 outputs). The training was followed up with a semi-structured interview. If deemed necessary by the researcher, the participant was asked to repeat previously completed tasks. Each instruction was marked as "pass" by the study supervisor who constantly monitored the participant's actions on the accompanying screen. During this process, the researcher provided oral instructions upon completion of a task, and whenever the participants requested assistance.

To minimise fatigue, participants were given a break of approximately 15 min after training and before the main experiment commenced. After training, the participants were asked to perform a task within the VR environment without a time constraint. The main task had an identical form compared to the training: *identify patterns in a multidimensional dataset of 26 input parameters and 3 output criteria*, together with the suggested approach to start developing the scatter plot using the criteria dimensions. We adopted a think-aloud

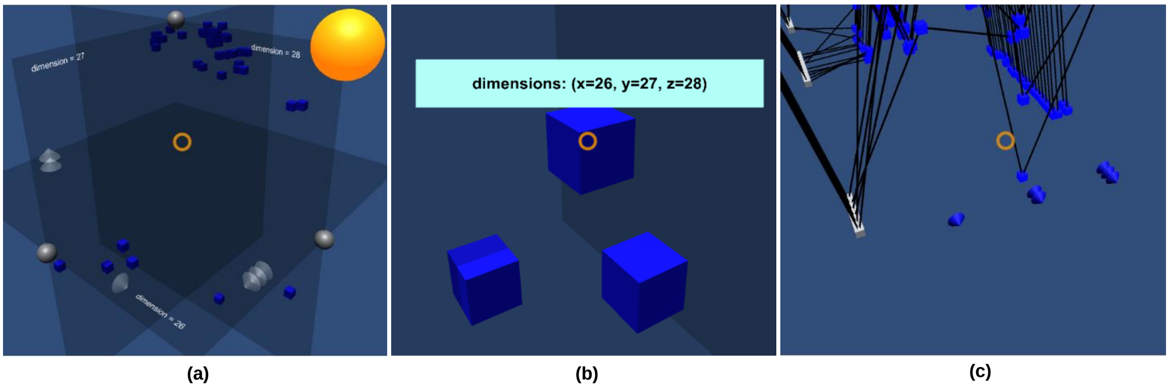


Fig. 5.4 (a) The 3D scatter plot generated from up to three dimensions selected on the IPCP. (b) A few volumetric data points (blue, unit-sized cubes) on the scatter plot. When the user gazes over the cube a text box with additional information pops up. The text box automatically follows the user's gaze. (c) The selected dimensions are colour-coded and marked on the IPCP with a number of cones. The design of these 3D floating scatter plots was adopted from Tadeja et al. [156, 150, 157].

protocol and requested participants to discuss what, why and how they are trying to achieve the task as it progressed. The supervisor, who continuously monitored the experiment, could at any moment inquire for further clarifications to gain more insight into the participant's actions and thought processes. In particular, for first-time gamepad users, an inquiry on whether the participant is stuck allowed the supervisor to provide guidance, for example, information about which button to click to invoke a specific action. This is important as the controller is not visible through the VR headset. Moreover, after running two participants (P1-P2), we decided to instruct the remaining participants (P3-P11) that a pattern is constituted by at minimum a group of four points. Audio and video were captured during the study for later analysis.

5.2.3 Participants Background and Expertise.

All the participants reported having extensive experience and background in the STEM (science, technology, engineering, and mathematics) fields in either industrial or academic settings or both.

As the IPCP system was updated after each study, the participants were recruited in three separate groups. As such, P1-P7 were recruited to carry out the first qualitative study. Next, P8-P9 were the volunteers participating in the second study, after the first round of IPCP upgrades. Finally, P10-P11 took part in the experiment with the larger dataset.

Participant 1 (P1): reported approximately six years of experience in engineering at undergraduate and graduate levels and two years spent in the automotive industry. He was not previously exposed to the PCP and had only very limited experience with VR.

Participant 2 (P2): reported less than three months of industrial experience. However, he has a background in aerospace engineering and engineering design spanning over nine years in total. He also considered himself novice with regard to VR with even lesser familiarity with PCP.

Participant 3 (P3): spent seven years in an academic setting and almost a year in the industry. She reported to have not previously used any VR headset. However, she is an expert on the PCP technique and its applications in design.

Participant 4 (P4): reported sixteen years of academic experience and three years of prior work in the industry. He is also an expert on PCP methods applied in modelling of the design processes.

Participant 5 (P5): has twenty one years of academic (15 years) and industrial (6 years) experience in Computer Science and software engineering. She also reported basic awareness with respect to VR and extensive exposure to PCP.

Participant 6 (P6): has a background in Artificial Intelligence and Computer Science (9 years). During that time, he has also worked for three years with the PCP and multidimensional data visualisation methods.

Participant 7 (P7): reported that he has an academic background in mathematics and computational fluid dynamics (8 years). However, he was a novice in terms of prior familiarity with VR and PCP.

Participant 8 (P8): was a doctoral student working on the application of analytical methods such us subspace-based dimension reduction to manufacturing deviations in turbomachinery components. He also self-reported having some limited experience with the VR and 2D PCP.

Participant 9 (P9): self-reported an intermediate level of familiarity concerning both 2D PCP and VR. This participant was also an expert in the field of mechanical and manufacturing engineering with vast experience of multi-objective optimisation in academia and industry.

Participant 10 (P10): reported having some limited prior exposure to VR and advanced knowledge about the 2D PCP. He used the 2D PCP for conducting research into his master's thesis in aerospace engineering. He also holds an undergraduate degree in the same area. In addition, he also recognised the dataset in which he looked for the patterns, as he used the same data in his prior research involving 2D PCP. However, he reported not spending a substantial amount of time on the analysis of this data, and he did not previously try to specifically identify the contained patterns.

Participant 11 (P11) self-reported to never before using the VR headset and to have some limited prior exposure to 2D PCP. He also reported to have undergraduate and postgraduate degrees in mechanical and aeronautical engineering and is currently in his first year of doctoral studies in aerospace engineering.

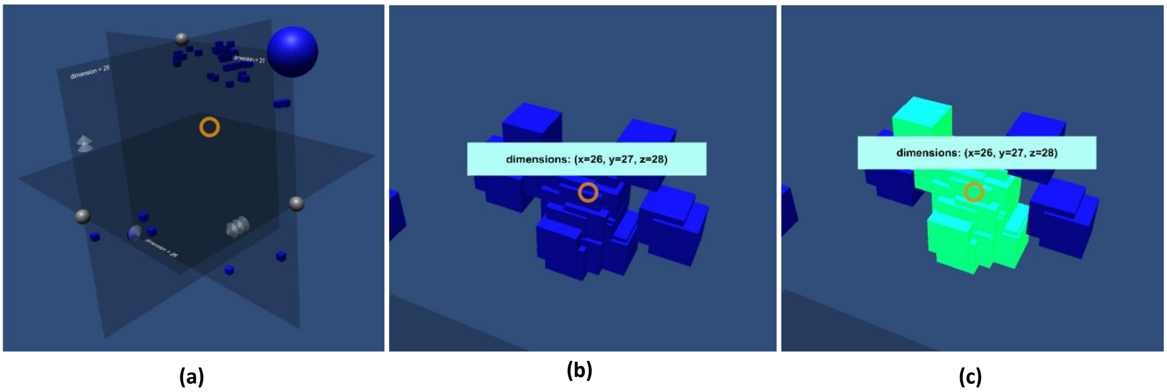

dimensions: (x=26, y=27, z=28)
dimensions: (x=26, y=27, z=28)
(a)
(b)
(c)

Fig. 5.5 (a) The 3D scatter plots [156, 157] provides built-in simplistic algorithm for automatic clustering of overlapping data points (b-c). If the volumetric data markers (i.e. unit-size cubes) overlap with each other (b) to a certain, predefined degree, the selection of one of them will automatically select the entire cluster as shown in (c). Moreover, these selections will be simultaneously reflected on the other visualisation elements, i.e. the IPCP and other scatter plots.

5.2.4 Identifying Patterns

Kipouros et al. [75] provided an analysis that discovered two patterns of importance, shown in Fig. 5.2 (b). As the study involved a real-world dataset, participants were instructed to note that some variability was to be expected in the data. However, they had to decide independently on what level of variability would be acceptable. Only two participants (P2 and P5) failed to recognise most of the data points belonging to the two groups. In three cases, the participants changed their initial pattern candidates by selecting or deselecting elements originally grouped, thus creating subsets or partial repetitions of the previous selection (P1, P4, and P6). The majority of the participants identified almost the complete pattern A (with completion ratios for P3:9/11, P4:8/11, P6:9/11 and P7:6/11) or B (with completion ratios for P1:16/19, P3:16/19, P4:15/19, P6:19/19 and P7:15/19). These differences most likely come from limited knowledge of the area from which the dataset was sourced and, as such, the allowed levels of noise between the respective values.

5.2.5 Observed Participant Behaviour

The results of the study involving first seven participants (P1-P7) suggest that the time needed for the analysis was correlated with prior experience with PCP, that is, the more expertise the participant had, the longer it took to finish the training session. Simultaneously, experts took slightly less time to complete the requirements of the main study. However, this data should be interpreted with caution. The participants did not have a clear cut-off point. For example, the study only finished once all the potential patterns were found. There was also an age difference between the two groups; novice participants were in their twenties (P1, P2, P7) whereas experts were in their thirties (P3, P4, P6) or forties (P5). Furthermore, the experts had little or no familiarity with VR or the controller, which put them at a disadvantage compared with younger participants, for instance, P7, who reported extensive experience with this type of gamepad. This may have impacted their approach used during the study.

All the participants quickly became fluent in their chosen movement and manipulation strategies and did not report any comments or suggestions linked to these interaction methods.

None of the participants generated more than a single scatter plot comprised of different dimensions even though they were informed about this possibility existed. Such behaviour, however, may have been observed because only three criteria were present in the dataset. Hence, all of them could be mapped onto a single scatter plot. Moreover, the training phase also exposed participants to the methodology of exploring the data mainly through the outputs' scatter plot which was also used by Kipouros et al. [75] in their exploration of this dataset. However, some of the participants—P1, P2, and P6—reflected on the fact that having more than a single scatter plot may be confusing.

We observed that participants P4 and P6 chose to re-position the scatter plot closer to themselves, or the particular part of the main plot, instead of moving towards the scatter plot floating above the centre of the IPCP plot. This behaviour would most likely change if more data items would be visualised at once. Therefore, the movement would take considerably more time, and the perspective would render the distant selection of the elements at the scatter plot almost impossible.

Only participants P2 and P6 chose to rotate the scatter plot. However, we observed that for all participants the exact mapping of the dimensions to the 3D projection was not critical and this was also confirmed later on during the final evaluation study. The participants were selecting clusters on the scatter plot irrespective of the current rotation of the projection. One participant (P3) was constantly selecting the same, largest cluster, instead of exploring other groups or individual points.

5.2.6 Design Principles

From the technical constraints of what is achievable in VR, expert feedback [75], and observations from our study we distilled six design guidelines:

D1—Immersive PCP visualisation: The IPCP design should leverage the benefits bestowed by the immersive, spatial environment and remain both useful and compelling to the user.

D2—Pattern candidate identification: The user must be able to identify candidates, i.e. a group of data items, as possible patterns within the dataset.

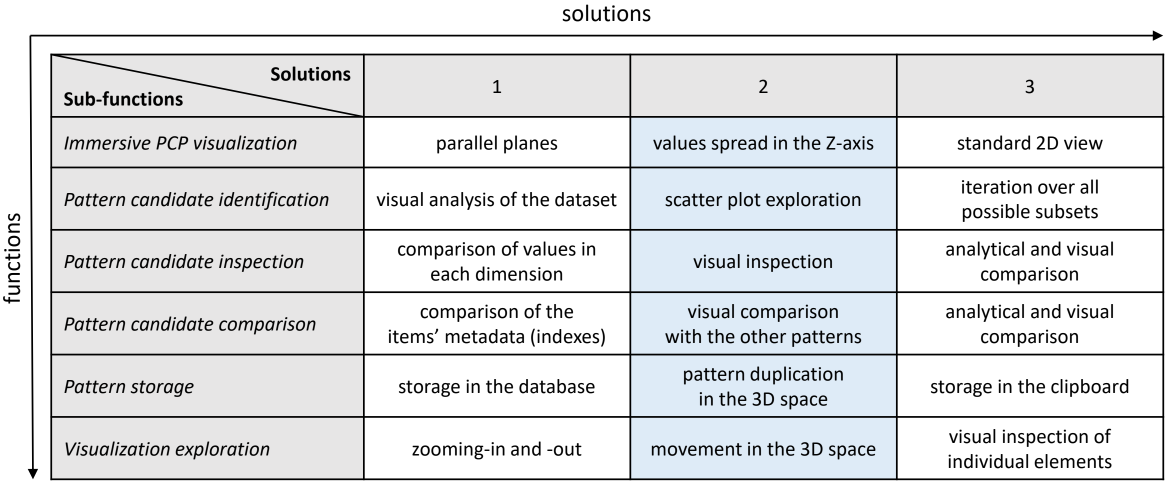
D3—Pattern candidate inspection: The user has to be able to inspect candidates for potential patterns within the dataset to decide if the particular group of data items forms a pattern.

D4—Pattern candidate comparison: To avoid repeated selections of the same pattern, the user has to be able to effectively compare a group of data items that could potentially form a pattern but also be able to compare such group with previously discovered patterns.

D5—Pattern storage: The possibility of saving and retrieving on-demand information about discovered patterns is necessary for the users to be able to differentiate between potential new candidates and previously discovered patterns.

D6—visualisation exploration. The system should support exploratory visualisation by providing mechanisms for zooming and exploring individual visualisation elements.

5.2.7 Task Analysis


solutions
functions

Solutions Sub-functions	1	2	3
Immersive PCP visualization	parallel planes	values spread in the Z-axis	standard 2D view
Pattern candidate identification	visual analysis of the dataset	scatter plot exploration	iteration over all possible subsets
Pattern candidate inspection	comparison of values in each dimension	visual inspection	analytical and visual comparison
Pattern candidate comparison	comparison of the items' metadata (indexes)	visual comparison with the other patterns	analytical and visual comparison
Pattern storage	storage in the database	pattern duplication in the 3D space	storage in the clipboard
Visualization exploration	zooming-in and -out	movement in the 3D space	visual inspection of individual elements

Fig. 5.6 Functions captured with the help of FAST technique were translated to the function carriers and presented in a form of morphological chart [115]. The 2nd column (in light blue) contains the actual solution that has been implemented in our system.

We performed a task analysis by first identifying a suitable user interface workflow guided by the previously described user study. Each task is broken down into functions carried out either in series or in parallel. Thereafter, each function was translated into a function carrier (solution) by considering the requirements, technical constraints, and data from the user study. This analysis was carried out by first modelling the function structures with the help of the *Function Analysis Systems Technique* (FAST) [134]. The results of this analysis was split into sub-parts presented alongside the individual design requirements. The translation of functions to function carriers is shown in the conceptual design tables on Fig. 5.6. For each key requirement (first column on the left) three alternative solutions (columns marked 1-3) were presented. For the benefit of the reader, the selected design solutions are listed in the 2$^{\text{nd}}$ column. The decisions were made using information gained through the formative user study in the form of observation and user comments and discussion. The list of these tasks include: (D1) *visualise data using IPCP*; (D2) *pattern identification*; (D2*) *scatter plot generation (augments the D2)*; (D3) *inspection of the selected data items*; (D4) *comparison of the new candidate to other patterns*; (D5) *storage of the identified patterns*; (D6) *exploration of the visualisation*. The diagrams flow from left to right; the branching should be understood as two overlapping possibilities that can be carried out together or augment each other.

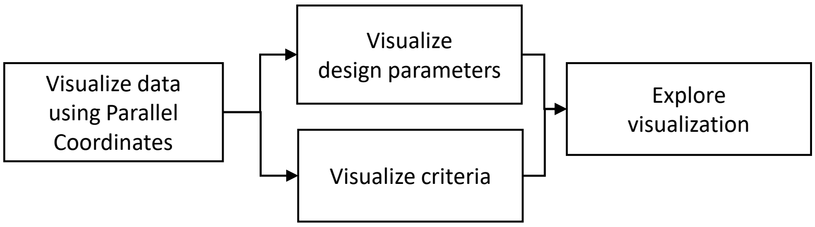

Visualize data using Parallel Coordinates
Visualize design parameters
Visualize criteria
Explore visualization

Fig. 5.7 D1: Visualise data using immersive PCP (IPCP).

D1—visualise data using IPCP: This task can be implemented in many variants in both 2D and 3D settings. For instance, one can use parallel planes [15, 68] to visualise the parallel axes. Instead, in our case, each data item is essentially a series of connected, interactive, cubes spaced in equal intervals over the X-axis. Identically to 2D PCP, the Y-component denotes the value for each dimension. However, as we are spreading the points also in the third dimension, the cube's position in Z-axis signals the data item indexing with the first element starting with the Z-component equal to zero and progressing towards the positive direction of the Z-axis (see Fig. 5.1). As opposite to other 3D PCP implementations, dropping the parallel planes limits the potential effects of occlusion and allows users to gain a birds-eye overview of the dataset by positioning themselves above the IPCP.

None of the participants reported confusion concerning the way the system visualised PCP. Moreover, all of them quickly understood that the interactive cubes indicate the numerical values in each dimension and that a group of line-connected cubes constitutes a single data item.

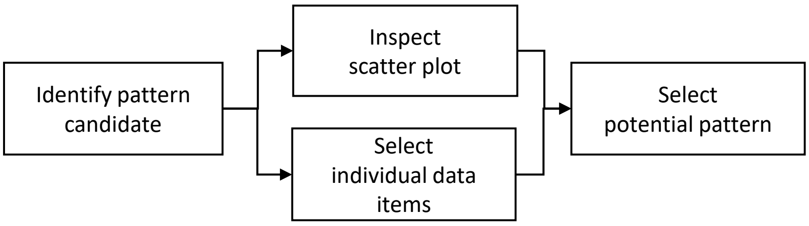


Fig. 5.8 D2: Pattern identification.

D2—Pattern identification: This task can be carried out in two ways: via exploration of the IPCP and visual inspection of a subset of selected data items, or with the scatter plot, as it was strongly suggested to the participants during the user study. However, we observed that the participants preferred a mixture of both approaches, that is, in many cases, a participant augmented a subset of data items selected with the help of the scatter plot by adding or removing selected data items individually from the PCP visualisation.

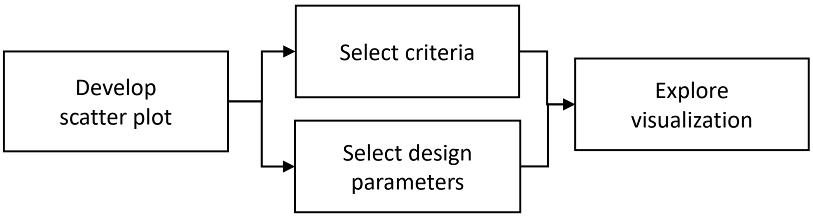


Fig. 5.9 D2*: Scatter plot generation.

D2*—Scatter plot generation: This task provides a feasible approach to pattern identification, especially when handling a large volume of data visualised with IPCP. The IPCP is augmented with the help of scatter plots [59, 75] generated from values in one, two or three dimensions across all the data items. Hence, we gave the participants the ability to create up to five different 3D scatter plots. We also advised our participants to start by selecting the criteria to develop the scatter plot and see if they can use this aid to identify clusters of data points on the scatter plot to help them efficiently select a group of data items that can potentially form a pattern.

The scatter plots were augmented (see Tadeja et al. [156, 157]) with additional functionality, such as the ability to apply rotation in 90° about a chosen axis. In addition, they could be moved into any position in 3D space.

Based on the participants' comments we decided to contextualise the data points and scatter plot axes by providing textual labels, as seen in Fig. 5.4 (a–b). The labels contained information of which dimensions, that is, which Y-values, were mapped to which coordinates on the scatter plots. However, the two expert participants suggested a re-evaluation of this design, as they found the labels not to be very helpful.

If two or more values in a particular dimension are identical within the floating-point precision of their coordinates, they will be presented as two entirely overlapping cubes on the scatter plot. This effect would interfere with the ray-tracing used for gaze tracking as the game engine has no means to decide with which overlapping object the user wants to interact with. The analogous situation may also happen in the 2D PCP. In such a case, only a single point will be visible on the scatter plot, the selection of which will highlight multiple corresponding items on the IPCP as those points would be automatically clustered together.

The simplistic, naïve clustering algorithm (see Fig. 5.5) relying on the spatial distance of the cubes' centres, was nonetheless quite effective. Certain restrictions, such as a difficulty of differentiating with which object users want to interact with if they are entirely overlapping, cannot be easily solved. We conjecture that different clustering algorithms for multidimensional data, such as one proposed by Wegman et al. [174] might be applied here. One way to minimise occlusion and cluttering effects in PCP is to use the density information of the data. This is shown in Heinrich et al. [55] and also in Artero et al. [7], where they proposed algorithms that use frequency and density information to decrease the overall number of data points that are shown to the user in PCP. Moreover, giving the user the possibility of zooming in and out of the cluster internal structure may prove to be very convenient in such situations.

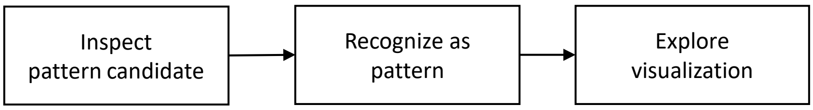


Fig. 5.10 D3: Inspection of the pattern.

D3—Inspection of the selected data items: This task is made possible through visual inspection of the selected items by either rendering not-selected points temporally invisible or by making a copy of the subset and placing it in a chosen part of 3D space. In both cases, participants usually first tried to gain an overview of the entire selection by positioning themselves (i.e., zooming out) in front of the subset and then compare and decided if particular data items constitute a pattern.

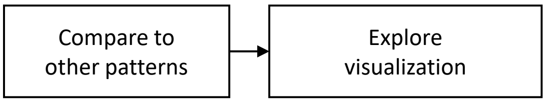


Fig. 5.11 D4: Patterns comparison.

D4—Comparison of the new candidate to other patterns: This task is achievable by placing candidates near each other in 3D space and then visually inspecting possible overlaps. However, one of the participants mentioned that having some analytical aid could be helpful, especially if the data items are confounded by many dimensions. This participant also suggested using colour-coding to provide additional information on how particular subsets potentially overlap with each other.

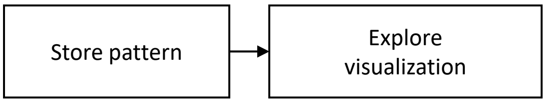


Fig. 5.12 D5: Pattern storage.

D5—Storage of the identified patterns: The system automatically logs a list of data items selected by the user into a file. Second, the sub-selection of data items can be placed and stored in any part of the 3D space throughout the entire duration of the visualisation. However, as the IPCP extends along the *Z*-axis it may be difficult to compare points that are placed on the opposite sides from each other.

In addition, participants also remarked that a combination of both of these methods could potentially be even more beneficial. The logged data can be made re-callable, visualised on-demand to assist the user in understanding, for example, if, and to what extent, patterns, overlap between each other by additional colour-coding or temporal shape change.

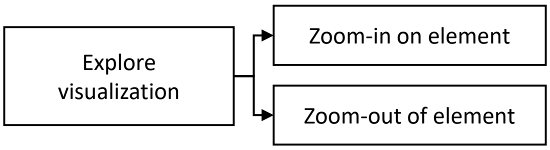


Fig. 5.13 D6: Visualisation exploration.

D6—Exploration of the visualisation: The movement and manipulation methods allow the user to zoom-in or zoom-out on any visualisation element, which, as reported by the participants, was easy to use with either Xbox Controller or Oculus Touch (see Fig. 5.3).

Based on the gathered qualitative data, all the participants swiftly understood the proposed movement and manipulation techniques. However, participants unfamiliar with the controller had problems remembering which button activated which action. This problem can be rectified by several strategies. First, the training can be extended. Second, the feature set can be limited. Third, the feature set can be initially limited and gradually expanded as the user gains proficiency in the interface. This third strategy is an example of a training wheel interface [18]. The ideal solution is probably a combination. In the short-term, extending training is preferable as PCP, even in its 2D version, demands intensive practice to gain fluency in their usage [138]. Another possible option would be to use a different VR headset that supports spatial tracking of the wearer. In such a case, instead of using a controller for movement and manoeuvring the user can move around in the real world which would be instantly reflected in the VR environment. This would decrease the number of required interaction methods, although it would raise other issues, such as the increased complexity in tracking set-up and space demands for the system.

5.3 Verification, Evaluation and Final Design

5.3.1 Interface Verification

To verify if our system offers all the functionality necessary to fulfil the primary user requirements we probed the usability of the system using the widely adopted Nielsen's heuristics [102, 101].

- *Visibility of system status*: The system constantly informs the user which visualisation elements are in a selected state on both the IPCP (see Fig. 5.1) and on the scatter plots (see Fig. 5.4 and Fig. 5.5). The selected objects remain highlighted (in light green or orange) up to their deselection or when the user invoked visualisation reset. This provides the user with continuous feedback about the state of all individual elements. Moreover, the system supports *linking & brushing* interaction techniques by automatically mapping the current state of each of the visualisation elements, i.e. scatter plots and the IPCP onto each other. Furthermore, if the user reaches the limit of possible duplicates, the system will disallow this operation and signal this to the user by playing a short sound.

- *Match between the system and the real-world*: Both the 3D scatter plots and the cross-hair based gaze-tracking (see Tadeja et al. [156, 157]) should be familiar to the majority of users. The use of the cross-hair to represent the user's gaze point and the focus the user's attention on an object directly in front of where the user is looking at the moment was implemented based on a well-known real-world concept [166]. This, in turn, allows the system to optimise its resources to provided immediate feedback and response to the user's actions.

- *User control and freedom*: Similarly to Tadeja et al. [156, 150, 157], the user can reset (see button [*R*] in Fig. 5.3) the visualisation to its original state which will preserve the copied data items (see Fig. 5.1). However, due to efficiency reasons, these duplicates will no longer support any kind of interaction after the system reboots and remain in "a freeze" state based on where they were originally copied. To further foster user control and freedom—before the final evaluation study with the larger dataset—the system was extended to support the *undo & redo* operations i.e. the last selection of data item on the IPCP or data point on the scatter plot can be undone or redone if necessary by the simultaneous press of two *thumb-rest* buttons (one on each of the controllers, see button [*H*] in Fig. 5.3).

- *Consistency and standards*: Despite the recent rapid development and advances in the research, the VR technology is still in its early stage. As a consequence, the standards regarding the interaction techniques, visualisation guidelines or the overall design principles are not yet fully understood. We were trying to be consistent with well-known concepts, thus, for instance, the selection of the object requires a double-tap of the gamepad button instead of a single click similarly to what is necessary when selecting an object using a computer mouse.

- *Flexibility and efficiency of use*: The "accelerator" provided for the benefit of advanced users was the possibility to generate multiple scatter plots. Each of them could map up to three dimensions from the IPCP. Moreover, some users possessed previous experience with the Xbox Controllers (see Fig. 5.3), and therefore required substantially less time to acquaint themselves with the interaction techniques.

- *Error prevention*: The most common "slip-type" of an error related to user failing to focus his or her gaze on a chosen point is easily fixable with manoeuvring techniques which allow the user to zoom-in the object by just moving towards it. The scatter plots automatically deselect the rotation selector when another one is being selected (see Tadeja et al. [156, 157]) by the user. This prevents the system from confusion about which axis the scatter plot should be rotated. Moreover, the system will omit visualisation of the data items on the PCP and the scatter plots if there is a gap in values at the input dataset, hence preventing the user from working with broken or incomplete data.

- *Help users recognise, diagnose, and recover from errors*: The system uses the audio feedback to inform the user if no more duplicates of the selected data items can be generated immediately after such action was performed.
- *Recognition rather than recall*: The *linking & brushing* interaction techniques help to minimise the user's memory load by providing constant feedback about which operations have been previously executed by the user. Moreover, the user can easily duplicate a group of data items for further inspection and comparison with other such groups. This feature can be used, for instance, to keep track on which group of points were previously recognised by the user as a pattern.
- *Aesthetic and minimalist design*: The systems keeps the information presented to the user to a minimum. Only selected elements are highlighted (see selected: data items on Fig. 5.1, movement selector on Fig. 5.4(a) and cluster on Fig. 5.5(c)). Moreover, the gazed-locked textbox is only shown once the user's gaze is hovering over a data point on the scatter plot and automatically disappear otherwise. Thus, minimising the possible additional cluttering and occlusion effects simultaneously providing additional information of which dimensions were used to map into this points' rearrangement on the particular scatter plot.
- *Help and documentation*: As it is hard to find users experienced in both the PCP and VR the participants were trained in how to read the PCP and how to interact with our system through the gaze-tracking and the Xbox controller (P1–P7) or the Oculus Touch (P8–P9).

5.3.2 Evaluation

It is very challenging to meaningfully evaluate visual analytics tools in short-term lab-based studies. Traditional human-computer interaction evaluation methods, such as A/B testing, rely on the experimenter's ability to explicitly control all controllable and uncontrollable parameters of the system and the task as much as possible. However, in visual analytics, this parameter space is very large. For example, the data density, the nature of the task, characteristics of the dataset used in the test, and the experience and motivation of the participants, are all variables that are likely to affect the outcome of an A/B test. In addition, easy-to-measure dependent variables, such as task time, are only partially relevant as one of the primary objectives of a visual analytics tool is to provide additional *insight* rather than enabling users to make decisions faster.

As a consequence, we evaluated the system design qualitatively with two independent user experiments. We recruited two pairs of different surrogate expert users to participate in each study. These participants are referred to as P8 and P9 in case of the study involving the same dataset as in the previous study and P10 and P11 with the new, larger, more complex dataset. All four volunteers were also pre-screened with the complete version of the Ishihara's colour deficiency test [65].

Part I: 54 Data Items with 29 Dimensions per Element

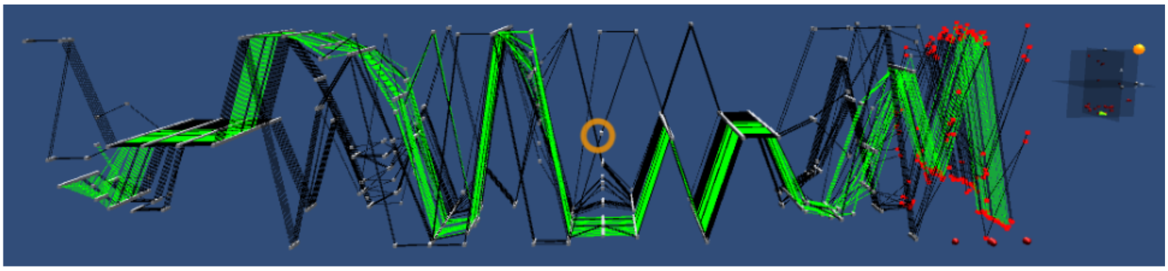


Fig. 5.14 The user's field of view of the 29-dimensional dataset of 54 equally optimum design configurations describing the Pareto front (26 design parameters and 3 objective functions).

Table 5.1 Group of indexes of data points identified as patterns by the participants P8 and P9. The two bottom rows contain the two patterns that were identified by Kipouros et al. [75] as can be seen in Fig. 5.2 in blue and green respectively. Here we report only the participants' best attempts to identify the two patterns.

Pattern candidate	P8	P9
1	33, 34, 35, 36, 38, 40, 41, 42, 43, 46, 49, 52	8, 9, 10, 12, 14, 15, 16, 17, 18, 27, 28
2	8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 21, 22, 23, 24, 25, 26, 27, 28, 30	8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 23, 24, 25, 26, 27, 28, 29, 30
Pattern A	32, 33, 34, 35, 36, 38, 39, 40, 41, 42, 43	
Pattern B	8, 9, 10, 11, 12, 13, 14, 15, 16, 17, 18, 23, 24, 25, 26, 27, 28, 29, 30	

To validate the system design, we used the same three-part procedure and dataset (see Fig. 5.1) as in the first study reported earlier. However, based on the previous participants' feedback, we skipped parts of the presentations related to having multiple 3D scatter plots, although this possibility was not removed from the IPCP. As such, the training phase was truncated. Overall, the results revealed that participant P8 managed to find the majority of the elements in both patterns, A with a 9/11 completion ratio and B with an 18/19 completion ratio, whereas the second participant P9 correctly identified only the entire pattern B with a 19/19 completion ratio.

Part II: 166 Data Items with 39 Dimensions per Element

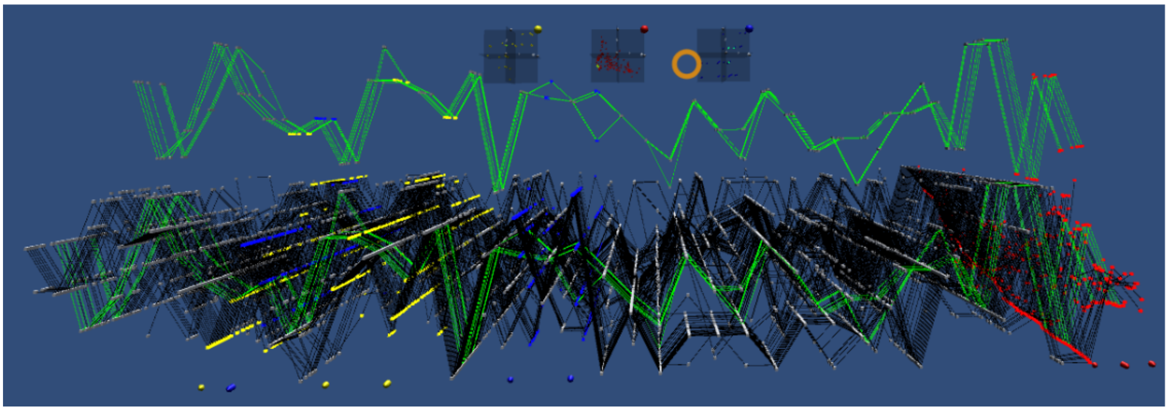


Fig. 5.15 The user's field of view of the 39-dimensional dataset (36 design parameters from FFD parameterisation and 3 design criteria: CP, DC60, and Swirl). The mapping i.e. the 3D scatter plot of the dimensions describing the design criteria i.e. *objective functions* is plotted (in red) in the middle scatter plot.

Table 5.2 Group of indexes of data points identified as patterns by the participants. The four bottom rows contain the four patterns that were identified by domain experts [32]. In the case of P11 we excluded repeated pattern selections.

Pattern candidate	P10	P11
1	93,96,102,103,105,119,120	44,46,49,56
2	5,9,12,25	92,95,98,106
3	13,15,17,19,20,21,23, 29,30,31,42,43	5,9,12,14,25,27
4	2,3,4,5,8,9,12,14,18,25,27	34,35,37,38
5	44,46,49,56,59	1,2,3,5,7,10,11, 12,14,18,22,25,27,59
6	90,92,95,98,104,106	110,114,116,117,121
7	-	2,3,7,18
Pattern C	44,45,46,48,49,51,56,59	
Pattern D	57,61,62,63,64	
Pattern E	13,15,17,21,29,30,31,34,35,37,38,42,43	
Pattern F	2,3,4,5,8,9,12,14,18,22,25,27	

After extending the system with additional capabilities, such as the support of basic *undo & redo* operations we reran our validation study with another two domain-experts (hereafter referred to as P10-P11). We also used the Oculus Touch instead of Xbox gamepad controller to facilitate the interaction (see Fig. 5.3).

We further investigated the feasibility of using IPCP for pattern identification in a substantially larger and more complex dataset. The user study was carried out over a new dataset containing 166 data items with 39 dimensions per element containing 4 patterns of interest identified by domain experts (see Fig. 5.15). This dataset describes the results of the optimisation study of S-Duct design (see Fig. 5.16), as presented in D‘Ambros et al. [32].

In addition, the near-misses in terms of selection were the main observable issues with our selection technique. To support the user's selection task, one can use a form of target acquisition aid, that in our case, has been loosely inspired by the *Bubble Cursor* proposed by Grossman et al. [49]. We leveraged the features natively supported by the Unity game engine by on-the-fly manipulation of the size of the *sphere collider* [165] component attached to the interactive objects (cubes) in the IPCP. The size of the collider, which acts as a target for the raytracer [166], is dynamically adjusted based on the size of cross-hair (see the orange circle in Fig. 5.15) which in turn is dynamically scaled with respect to the user's distance from the target objects [166]. Initially, we set the collider's radius as $\sqrt{3}/2 \approx 0.866$ to encompass the entire cube in the sphere. The further away the user is from the object, the larger the collider becomes.

P10 found 5/8 of pattern C, 9/13 of pattern E and almost all pattern F: 11/12. P11 found 4/8 of pattern C, 4/13 of pattern E and 9/12 of pattern F in his best attempt. However, none of the two participants found pattern D. They did, however, identify other potential patterns that were not initially found by the original researchers, which is a concrete demonstration of the utility of the system in enabling users to explore and identify previously unknown patterns in the dataset. Interestingly, some overlapping results were reported by both participants, as can be seen in Table 5.2, namely pattern candidate 6 for P10 and pattern candidate 2 for P11. Moreover, both participants identified additional groups of points that in their opinion constituted patterns not originally identified by the experts in D‘Ambros et al. [32]. As mentioned before, in the case of visual analytics, we aim to provide additional insight about the dataset, hence we were interested in the new candidates for potential patterns found by the two participants (P10 and P11). To do this, we requested a domain-expert to re-analyse and comment on the new pattern candidates identified. After inspection and preliminary analysis, the expert concluded that indeed these are different patterns sharing some common characteristics with those identified previously and as such they are complementary. However, the domain expert commented that they are not as rich as the dominant four patterns previously identified, i.e. C, D, E and F. Nonetheless, this is concrete evidence that VR analytics enable users to detect new patterns.

5.3.3 Design Decisions

Here we summarise the final design decisions refined after we ran additional experimental studies with domain experts over two separate datasets (see Fig. 5.14 and Fig. 5.15) varying in both number of data items and dimensions per data item as well. Moreover, based on the participants observed behaviour and their comments we also identified a number of potential improvements.

D1—Immersive PCP visualisation: We have developed immersive PCP (IPCP) as a software system running within the Unity [165] game engine. Our approach represents a 3D version of regular 2D PCP but with values of each dimension denoted as volumetric, interactive objects, that is, cubes spread in the Z-axis instead of using parallel planes as regular 3D PCP usually does. However, this way of presenting data has its own challenges. For instance, to clearly see the distant objects with respect to the user's current position, it was necessary to provide adequate means of close inspection of such regions. Here, we give the user the possibility to freely move and manoeuvre in the 3D space using the controllers (Xbox gamepad in first and Oculus Touches in the following two observational studies). Such approach has additional advantages, especially when we coupled the IPCP visualisation with the CAD models of the respective designs (see Fig. 5.16) which now can be inspected not only from outside but also from within as well (see Fig. 5.16(c)).

D2—Pattern candidate identification: To identify and highlight a group of data items that can potentially form a pattern, the user either selects data items one-by-one or by generating a 3D scatter plot from up to three dimensions. This data is automatically clustered based on the distance between the data points (see Fig. 5.5) and the selection of such data points belonging to a cluster simultaneously highlights points belonging to this cluster on the scatter plot and the IPCP alike. The larger the data set and subsequent IPCP visualisation, the more the scatter plots seem to aid the user in the task of finding patterns. Here, we noted that the actual orientation of the scatter plot's local axes is not as important as providing the user with a clear way of distinguishing which points on the scatter plot potentially form a cluster. We observed that even a simple clustering algorithm, however relying on the user's intuitive feel of the grouping of volumetric points placed in close vicinity can be surprisingly effective. Hence, potential improvements could be threefold. First, other clustering algorithms can be studied here such us *Ordering Points to Identify Cluster Structure* (OPTICS) [5] or *Density-Based Spatial Clustering of Applications with Noise* (DBSCAN) [36] as they can, by design, handle noisy data, e.g. containing almost constant dimensional values across all the data items, such as the one used in our experiments. Second, different methods of the 3D scatter plot manipulation can be used. For instance, instead of rotating the scatter plot along one of the three main axes, it could be rotated about any vector selected by the user. These would potentially help the user to swiftly obtain an overview of the entire data present at the scatter plot. Third, to avoid repeated selection of the same cluster, and subsequent repetition of the same group of data items on the IPCP, we could additionally colour-code or otherwise previously selected and inspected clusters. Furthermore, additional information could be given in a form of textual labels attached to the data points instead of current axes mapping, such as the scatter plot's coordinates or number of points within selected cluster which could be more informative to users.

D3—Pattern candidate inspection: To inspect the pattern candidates, the user can temporally disable the unselected data items of the IPCP and use the manipulation and movement capabilities in the 3D space to verify if this particular group of data items forms a pattern. Moreover, if a particular data item is upon inspection deemed by the user to not be part of this pattern, it can be easily diselected as well. This is especially useful the more data items are contained within the dataset. As the data items will be spread in the *Z*-axis (see Fig. 5.14 and Fig. 5.15) the potential occlusion effects are more persistent with the growing number of data items. Here, the ability to toggle between the view with all or only currently selected data items has proven to be very useful.

D4—Pattern candidate comparison: To compare the currently selected group of data items with previously recognised patterns, the user can make a copy of these data items, i.e., the pattern candidate, and place them in a part of 3D space where the other patterns have been stored. The user may then visually compare them to decide if a new pattern has been found. Here, the system should provide the user with the "snapping" mechanism that would automatically align the subsets of the data items that the user wants to compare with each other. To some extent, the participants were trying intuitively to obtain a similar effect by placing the duplicates of previously found patterns next to each other in the 3D space.

D5—Pattern storage: Previously discovered patterns of data items can be stored freely in any location within 3D space. Here, even though we could keep the "frozen" data items as non-interactive for optimisation reasons, we could colour-code the data items overlapping across the previously found patterns and the new group of points to inform the user to what extent they share the same subset of data.

D6—Visualisation exploration: Exploration of the entire visualisation with all of its elements is made possible through the use of gaze-tracking, head-tracking and movement assisted by the controllers. Here, we observed that all our participants swiftly gain fluency in how to move around the virtual space. We can also leverage the tracking of the user in a real space. However, in our case, the IPCP, – pending on the input data size – ca be potentially vastly extended in at least the *XZ*-plane parallel to the user actual ground-plane. As such, it could be unfeasible for the user to walk and see the entire visualisation, hence new interaction and transportation methods would have to be included.

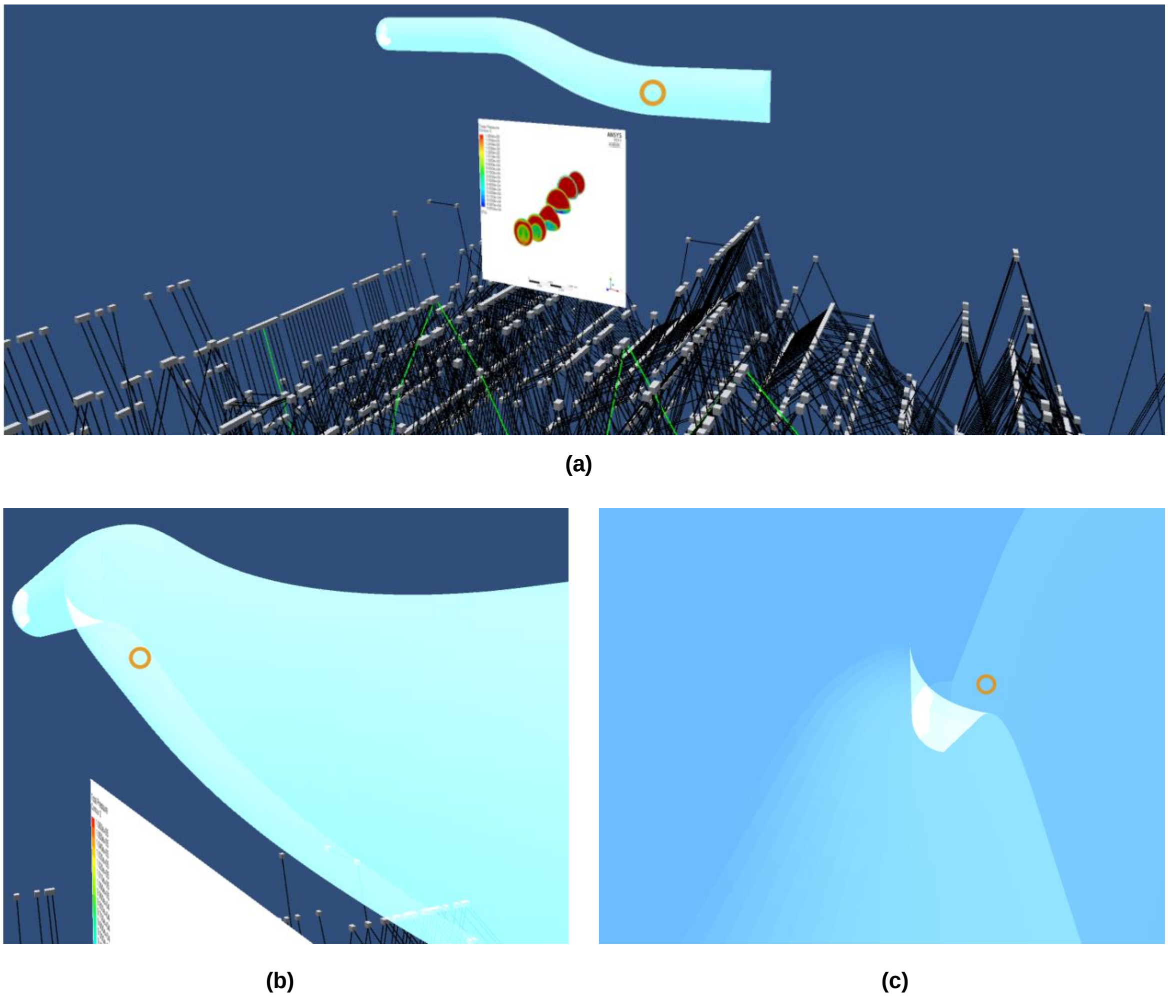


Fig. 5.16 After the selection of a data item on the IPCP (a) the user is shown and can inspect additional information associated with that particular design (a) the semitransparent s-duct CAD geometry and accompanying 2D plot representing the flow going through the s-duct. (b) Shows the same CAD geometry closely inspected by the user from one of its sides, and (b) from within the model itself.

5.4 Visualisation Extensions

After analysing both the qualitative and quantitative feedback together with the participants' comments, we have decided to extend our system with additional features. The system contains the visualisations of associated 3D models of the full 3D S-Duct geometries and 2D flow solutions, an example of which can be seen in Fig. 5.16. In the future we are also planning to give the users the possibility of overlaying the various S-Duct geometries on top of each other to closely inspect and identify potential nuances in the difference between the shapes as demonstrated in the AeroVR system presented in Tadeja et al. [156, 157] and upon which this visualisation is built. This allows the users to promptly verify their pattern selections and compare between the S-Duct shapes associated with selected data items. This extension also fosters the connectivity with commercial software and simulation packages that are widely used in the industry.

In addition, the tool automatically provides some form of a collaborative design environment. The first-person view of the headset wearer can also be simultaneously streamed on the PC display as well. As a consequence, multiple designers can look at the steps undertaken by the user and easily collaborate having a verbal dialogue between themselves and the person using the headset. This capability was observed by the authors while working on the design and verification of the IPCP tool presented in this thesis. The advantages of such setup will be further studied in a series of a more controlled user studies.

5.5 Discussion

A previous study conducted by Johansson et al. [68] indicated that 3D PCP was, in fact, worse than a conventional 2D version in both the metrics (i.e., response times and error rates). However, the study was conducted using a desktop computer connected to an external display and it is unclear whether the findings generalise to truly immersive environment.

The qualitative studies presented earlier in this Chapter show tentative evidence toward the possibility that immersing the user within PCP in a VR environment is, at the very least, *complementary* and may provide ways to *enhance* the user's ability to identify patterns. Therefore, it can be concluded that immersive 3D PCP can potentially be considered as a feasible way of visualising high-dimensional data. Most of the participants throughout our qualitative studies managed to identify the same patterns as recognised as interesting in Kipouros et al. [75], which can be regarded as the reference standard. Moreover, the two participants working with a substantially larger dataset in terms of its complexity and size, i.e. the number of dimensions per data item and the total number of data items, found similar patterns as those given in D‘Ambros et al. [32] and identified additional candidates for potential patterns. This demonstrates the potential for immersive VR in engineering design processes.

5.6 Conclusion

The main contribution of this Chapter is the task analysis carried out at the functional level and the translation of these functions into a solutions that allow the system to be realised. This scaffolding structure is useful in framing the findings from the preliminary qualitative user study in such a way that the outcomes of the user study could directly shape the refinement of the initial design.

The evaluation with a group of surrogate expert users, P8–P9 for the smaller and P10–P11 for the larger dataset, revealed that users were indeed capable of identifying patterns in both datasets. Moreover, the results of the carried user studies demonstrate that IPCP can be viable option for VR analytics especially in comparison to previous approaches discussed in Chapter 2. However, while this evaluation demonstrates the efficacy of the system and serves to provide some evidence to consider the potential merits of 3D PCP, these results have to be analysed with caution given the qualitative aspect of the findings. The sample size is very small and, as a consequence, the findings are highly sensitive to the individual participants' general ability to use novel immersive technology to analyse a demanding real-world multidimensional dataset. Future work includes conducting a statistically robust quantitative study that is able to tease out the merits and flows of immersive PCP visualisation compared with the status-quo 2D PCP visualisation. However, such studies are difficult to carry out due to the many controllable and uncontrollable parameters of the system and the experimental task, which makes the design of a robust and meaningful experimental design highly challenging. Prior work has reflected about this problem, both in general for evaluating visualisation tools [163, 169] and specifically for the problem of comparing 2D and 3D visualisations [79].

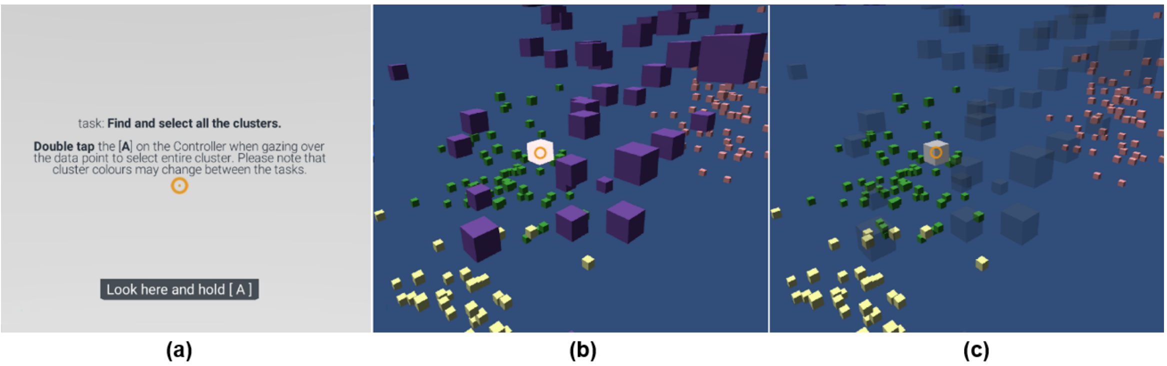
Chapter 6

Evaluation of Virtual Reality Analytics Interfaces

The central contribution in this Chapter is a generative process for synthesising datasets that allow designers and researchers to carry out iterative design and evaluation using successive A/B testing with the same set of participants. Normally this is problematic as the participants are going to learn the intricacies the dataset and thus the (unavoidable) learning of the dataset becomes an uncontrolled variable.

There are currently two methods to mitigate this problem. The first method is to use different datasets. However, if these datasets are natural datasets then there is a high risk the characteristics of the individual datasets will inadvertently become a variable that risks explaining a large proportion of the variability between different conditions. The second method is to change to a between-subjects design and recruit new participants for each successive iteration, which is expensive, reduces statistical power (as the variability of each individual is no longer controlled across all conditions and/or iterations), and prevents studying longer-term learning effects of the interaction techniques or VR tools themselves.

In here, we propose synthesising 3D datasets using a robust underlying process. This work will demonstrate that this approach leads to datasets that still allow researchers to detect significant differences for different tasks over a variety of metrics, while at the same time minimising the influences of the datasets themselves on the results. The process therefore allows repeated measures designs in which, for example, the parameters of an interaction technique can be manipulated, and the effects of such manipulation quantified, without having to be overly concerned of the influence of either participants learning a dataset or the intrinsic differences of individual datasets dominating as explanatory variables. The generative method allows the designer or researcher to quickly generate a sample of colour-coded clustered 3D data points of any desired size (number of clusters and number of data



task: **Find and select all the clusters.**

Double tap the [**A**] on the Controller when gazing over the data point to select entire cluster. Please note that cluster colours may change between the tasks.

Fig. 6.1 Interaction occurs through the gaze-tracking and double tapping the [A] button on the Xbox controller. (a) shows a screen shot of the dialogue scene with instructions of the *Task 1* as they are presented to the user in VR. (b) presents the gaze tracking i.e. orange cross-hair hovering over data point in a cluster (violet) which automatically highlights looked-at object informing user that it can be interacted with. (c) shows the same cluster selected (semitransparent) once the user double taps the [A] button on the controller while gazing over an interactive object.

points in clusters). Furthermore, the process allows parameter modifications to facilitate particular needs for an individual study. While this generative process specifically addresses clustering concerns, Bach et al. [9] reflect on the fact that 3D point-clouds can represent a number of different 3D visualisations, including, but not limited to, “3D-scatter-plots, specific space-time cubes, as well as bio-medical images” [9].

6.1 Approach

The difficulty in evaluating visualisation techniques is exacerbated in VR. All problems identified by, for example Isenberg et al.[64] still apply, yet VR in itself introduces several additional design variables, such as the navigation strategy, means of supporting spatial awareness, and effective direct and indirect manipulation techniques. Other classes of well-known problems include overplotting and occlusion [110]. Yet, to successfully navigate this design space, it will be necessary to empirically compare different interaction solutions. However, this raises a methodological problem in that the dataset itself is an important explanatory variable of user behaviour.

In here, this Chapter proposes a solution to this problem, which is relying on generating datasets based on a model. The generated datasets share similar fundamental characteristics and virtually any number of them can be generated in a matter of seconds. Depending on their needs, designers and researchers can change the design parameters, such as the number of clusters or their densities. To decide upon the clusters’ placements in 3D space, the process adopts a *random walk* (Brownian motion [34, 142] or Wiener process [176]). This model is widely used in physical chemistry, computational physics, stock market models and crystallography (see [175, 95]). A *random walk* is by its nature self-similar [24, 54], which is in itself a beneficial characteristic as many datasets also have some sort of fractal structure.

6.2 Generating Synthetic Clusters

The generative process can be split into four steps: (1) determine the position of the central point of each cluster in 3D space; (2) determine the size of each individual cluster; (3) generate samples of data around each clusters' central points; and (4) colour-code all individual clusters. Pseudo-code for both the generation process and colour-coding are shown in Algorithm 1 and Algorithm 2 respectively. The cluster's size and, indirectly, its spatial spread of individual data points influence if a newly generated cluster will be included within the dataset or regenerated again if it overlaps with other clusters above a certain threshold. As the last, non-mandatory step, all points are translated so the bounding sphere spanning along the entire dataset has its centre at the axes-origin, which is also where our testing framework described later in this Chapter initially places the user. We will now describe steps 1–4 in detail.

6.2.1 Cluster Placement in the 3D Space

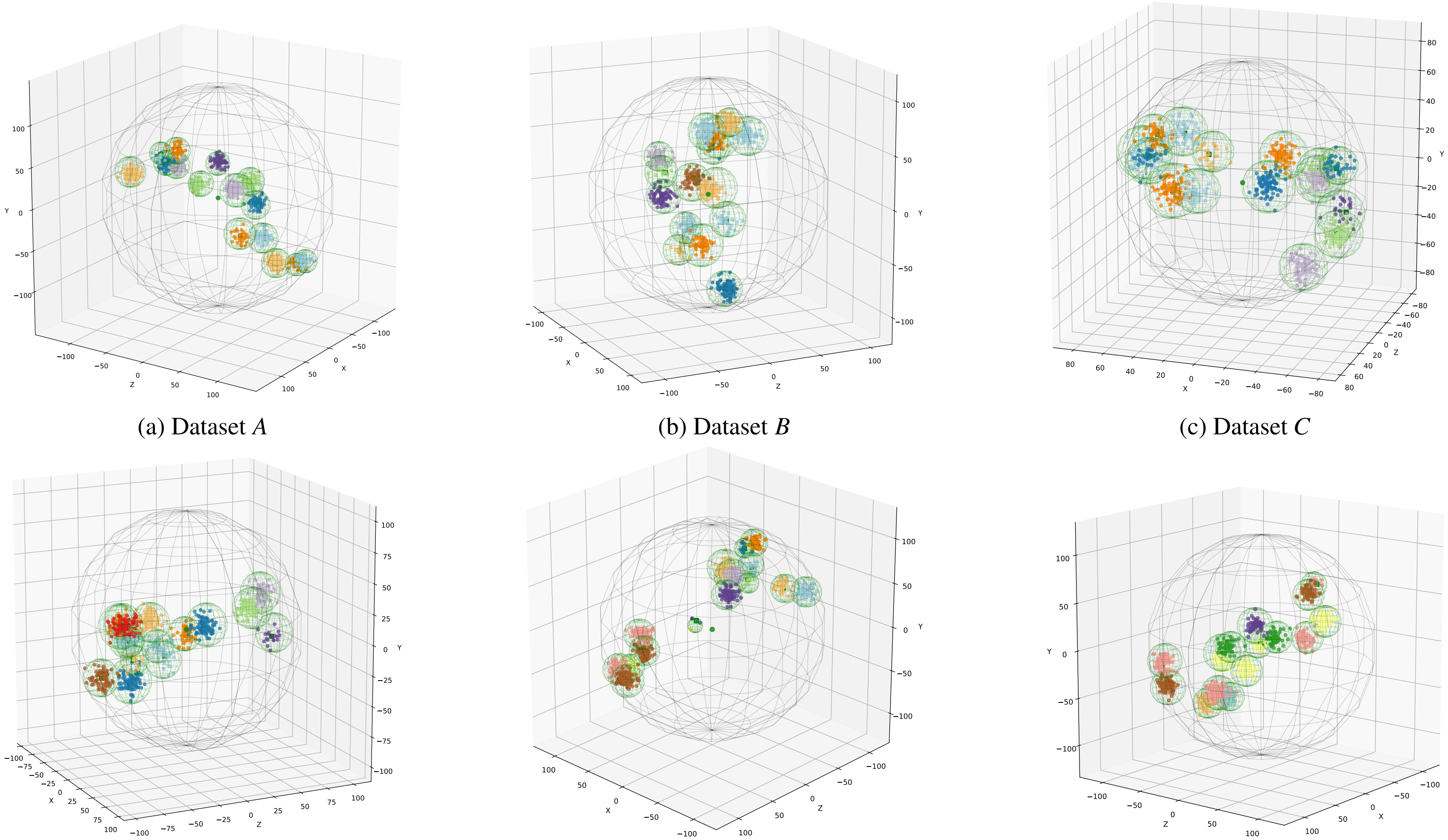
We first generate a 3D Brownian motion trail, which results in a set of $2N$ candidate points for possible cluster placements. We then draw cluster placements $P_1, P_2, \dots, P_N$ from this set of candidate points by successively sampling from a Poisson distribution with the rate parameter $\lambda = N$. If the candidate P_i was drawn previously, a new cluster is generated about this point. This cluster is either added to the dataset or it is disregarded due to it exceeding the overlapping threshold with preexisting clusters. In the case when the newly generated cluster is discarded, the candidate point necessitating this new cluster is disregarded as an optimisation step as it may provoke repeated overlap. Instead, once the candidate is disregarded, a new candidate P_i is generated in its place.

6.2.2 Cluster Size

To determine the cluster's size in terms of its total number of data points we sample from a power-law probability distribution $f(x,a) = ax^{a-1}$, with $a = 1{}^2\!/_3$, $x \in [0,1]$. The choice of a power-law is motivated by its scale-invariance and by how frequently many physical, biological and artificial systems generate this relationship. Once a cluster size is determined, we successively generate the desired number of points by sampling from a multivariate normal distribution $\mathcal{N}(\mu = P_i, \sigma^2)$.

6.2.3 Maximally Acceptable Overlap

There are many ways to define an overlap. For example, it can be determined by the total number of individual volumetric markers that occupy the same space. Another possibility is to consider a cluster as a chunk of space that is occupied by all its elements. Such a slice can be obtained by encapsulating it within a bounding body. Although not very precise, this method is not only conceptually and computationally simple but also commonly used in computer graphics [82] and it is the approach taken here. This approach does not generate clusters within the spheres but is using the bounding volumes calculated on top of the pregenerated set of 3D volumetric points. Such clusters can take on any shape due to their pseudo-random generation process. Using a bounding body may also be convenient when managing dynamic datasets where the overlap may have to be quickly recalculated during program execution.



(a) Dataset *A*

(b) Dataset *B*

(c) Dataset *C*

(d) Dataset *D*

(e) Dataset *E*

(f) Dataset *F*

Fig. 6.2 The figure shows the synthetic datasets $A - F$ respectively, each containing $N = 15$ clusters. The datasets are present from different angles, selected to empahsize the spatial distribution of each individual cluster. The figure also shows the bounding spheres of all individual clusters (in green) and the overall bounding sphere (in black) with its center (green dot).

We first calculate bounding spheres using Ritter's algorithm. [116]. We then define an overlap level $o_{i,j}$ of two clusters x_i and x_j as a function of the Euclidean distance between the centers of their respective bounding spheres $w_{i,j} = d(x_i, x_j)$ and the sum of their respective radii:

$$o_{i,j} = 2[1 - \frac{w_{i,j}}{(r_i + r_j)}]. \tag{6.1}$$

If $w_{i,j} = r_i + r_j$ (that is, $o_{i,j} = 0$), the clusters overlap in a single point, that is. their respective bounding spheres touch each other. If $o_{i,j} < 0$, the spheres are disjoint and if $o_{i,j} > 0$, the clusters overlap.

The values of $o_{i,j}$ are compared with thresholds $t_{i,j}$ to decide if the overlap is acceptable or not. Each $t_{i,j}$ is sampled from a power-law probability distribution with density $a = 1^2/_3$, $x \in [0,1]$. If any of the thresholds are exceeded, the cluster is disregarded. Optionally, the algorithm's decision can be assessed independently either by inspection or by automatically using any of the well known clustering algorithms [118, 183]. Since the data points are generated in clusters this information can be overlaid with clusters determined separately by a clustering algorithm to automatically estimate the percentage of points overlapping with the other clusters.

6.2.4 Cluster Colouring

Colouring clusters is a non-trivial task as factors, such as distance between clusters, their spatial distribution and the limited number of perceptually distinct colours, have to be considered. In addition, a further design constraint is the desire to reduce the risk of overlapping clusters with the same colour. We colour code the clusters according to a heuristic method that incorporates all of the above factors. The algorithm starts by assigning a colour Ω_0 to the first cluster located at x_0. Next, in each step, by using a greedy optimisation procedure, we choose the colour for one cluster located at x_i. First, for all colours $c_1, c_2, \ldots, c_K$ we calculate quality coefficients $A(c_k)$ defined as

$$A(c_k) = \sum_{j=1}^{i-1} w_{i,j}^{-2} [K - g(\Omega_j, c_k)], \tag{6.2}$$

where the weights $w_{n,i}$ are inverse and squared to ensure that the further away the clusters are, the smaller impact this distance will have on their color assignment, K is the number of

Table 6.1 The 2nd and 4th columns list indexes of selected colors and overlapping clusters respectively, whereas the 3rd one denotes their sizes.

Dataset	**Selected colors**	**Sizes** *(the largest and smallest are in bold)*	**Overlapping**
A	$\{0,6,0,7,7,1,8,2,9,2,8,1,7,0,6\}$	$\{66,\mathbf{99},69,32,30,62,94,94,64,92,88,42,44,\mathbf{14},98\}$	$\{0,3,12,13\}$
B	$\{0,6,0,7,1,8,2,9,6,0,7,0,1,6,11\}$	$\{87,78,85,85,18,62,\mathbf{15},73,\mathbf{99},73,62,32,95,20,74\}$	$\{3,4,9,10,13\}$
C	$\{0,6,7,1,0,7,0,7,1,8,1,2,8,9,2\}$	$\{66,30,50,33,75,95,45,71,\mathbf{97},75,39,77,86,\mathbf{27},80\}$	$\{10,9,2,3\}$
D	$\{0,6,7,0,7,6,0,11,1,0,5,1,8,2,9\}$	$\{37,92,46,22,58,19,22,73,\mathbf{96},71,94,94,65,84,\mathbf{18}\}$	$\{9,13,4,12\}$
E	$\{0,6,0,7,1,8,6,2,11,4,9,10,11,4,11\}$	$\{73,43,21,55,10,86,65,15,\mathbf{4},92,76,62,74,\mathbf{97},91\}$	$\{3,4,9,11,12,13,14\}$
F	$\{0,6,11,4,10,4,3,10,9,3,10,4,11,10,4\}$	$\{86,72,88,84,54,92,55,83,63,58,30,66,54,\mathbf{96},\mathbf{24}\}$	$\{9,10,4,6\}$

available colors and g is a *minimal color distance* between the clusters defined as:

$$g(c_i,c_j) = \min(|c_i - c_j|, M - |c_i - c_j|). \tag{6.3}$$

In our case, c is an integer index that can be easily mapped to the respective colour from a set of $K = 12$ different colours downloaded from the *ColorBrewer2.0*[31]. The coefficient M is either equal to $M = K$ when K is even, or $M = K + 1$ otherwise. Subsequently, we pick $\operatorname{argmin}(A(c_i))$ as the desired colour. The whole operation has $O(N^2K)$ time complexity. For a reasonably small number of clusters, the weights w and colour distances g are calculated when needed. However, the method can be optimised by precomputing all of $w_{i,j}$ and $g(c_i,c_j)$.

6.3 Immersive Analyics Environment

We built a VR analytics environment to visualise the generated clusters and to be able to interact with them using the Unity game engine and additional freely available assets[107, 166].

An example of a visualisation of a scatter plot can be seen in Fig. 6.1, which features four distinctive clusters, with the violet clusters presented before (b) and after selection (c). Each cluster is constructed from a set of points visualised as volumetric objects.

Clusters are visualised by grouping points together in terms of their respective attributes, such as colours and placements in 3D space. However, the final assessment on whether a point belongs to a particular cluster or not is left to the user, and only once the point is selected the whole cluster instantly becomes translucent.

Movement in any direction in 3D space is achieved using an Xbox controller. Movement occurs with respect to the user's gaze and current position within the VR environment.

When the user hovers over a data point (see Fig. 6.1) with the cross-hair, the system automatically highlights the data point by instantly changing its colour to white. Once a user is gazing over any of the data points in a cluster, the cluster can be selected by double tapping the [A] button. This makes all the elements belonging to the selected cluster translucent, as suggested by Rekimoto et al.[72] (see Fig. 6.1). Choosing this selection method helps to decrease occlusion, as pointed out by Shneiderman[136].

The environment partially supports six of Shneiderman's et al. [138] seven basic information visualisation tasks (*details-on-demand task* is not implemented). Among the remaining six, the *overview task* and the *zoom task* are supported by the user's movement capabilities in 3D space. The *relate task* is part of the visualisation itself through a mixture of cluster elements' colour-coding and their placement in space. To a limited extent, the filter and history tasks are supported by keeping previously selected clusters translucent.

6.4 Validation

e validated the generative process in two experiments with identical designs. The validation had two objectives. First, to investigate whether the generative process resulted in suitable clusters for analytics design and evaluation. Second, to assess how much of the variability that can be explained by treating the dataset as an independent variable in a typical analytics experimental task.

Ideally the dataset variable can be treated as a controlled variable across several A/B studies, which then allows iterative A/B testing and fine-tuning of VR analytics interaction techniques without worrying about 1) the learning effect of using a specific dataset in every single condition; or 2) an undue noise effect arising from using datasets with widely different characteristics in every condition.

We find that, overall, the dataset does not result in significant differences at a weak significance level of $\alpha = 0.05$, which suggests designers and researchers can use the generative process described in here to keep generating new datasets for every within-subject condition and thereby eliminate the learning effect of using a specific dataset in every condition.

As we wanted to validate the generation method, we investigated the strength of the null hypothesis that there will be no noticeable significant performance differences between users for both the task type and dataset. While accepting the null hypothesis does not form conclusive evidence that it is true, failing to reject the null hypothesis at a significance level of $\alpha = 0.05$ (which is a weak significance level and therefore conservative in this case), indicates that for practical purposes of standard A/B testing, the dataset itself is unlikely to be the dominant explanatory variable of any experimental results. In addition, we also report effect sizes which help quantify how much of the variability is explained by treating the dataset as an independent variable.

To examine the sensitivity of the participant groups and datasets themselves, we split the study into two independent experiments that were later on analysed together.

6.4.1 Participants

For the first experiment we recruited 18 participants using opportunity-sampling. All of them were prescreened with a short version of the Ishihara's [65] colour deficiency test before commencing the experiment. Participants were within the age of 22–47, with the majority being under 30 years of age. Four of them were female and 14 were male.

For the second experiment we recruited 21 participants using opportunity-sampling. Half were female and half were male. The youngest participant was 22 years old and only three participants were above the age of 30.

6.4.2 Procedure

We used the generative process presented in here to generate six datasets for two identically designed experiments. Fig. 6.2 show *Matplotlib* [60] visualisations of the datasets A, B, C and D, E, F used in both experiments, respectively. The main characteristics of these datasets are listed in Tab. 6.1.

The two experiments were carried out in an identical fashion. Both were split into three series of three tasks and each series had a balanced order of the three tasks and used its own generated dataset (see Tab. 6.2).

After each cycle we asked participants to fill out a set of questionnaires, the NASA Task Load Index (NASA TLX) [53], the Simulator Sickness Questionnaire (SSQ) by Bouchard et al. [12] and originally developed by Kennedy et al. [74], and an English version of the Igroup Presence Questionnaire (IPQ) [61] administered through a web-based interface. We decided not to ask participants to fill out the forms after each individual task as this would significantly extend the time required to finish the experiment, which in turn would have an effect on a participant's levels of fatigue and overall performance. An analysis of these forms revealed slight increase in nausea and/or oculo-motor effects in the majority of the participants (13 in the first and 10 in the second study, respectively). However, in no case did a participant decide to stop the experiment or to directly report any moderate or severe symptoms.

Participants were orally briefed before the experiment and task-specific instructions and a written repetition of the oral brief were provided through the HMD before a participant begun a new task, as shown in Fig. 6.1. Participants were instructed to carry out the tasks as quickly and as accurately as possible.

6.4.3 Tasks

Participants were instructed to perform three tasks for all three datasets in each experiment. *T1*: Find and select all the clusters; *T2*: Find the smallest cluster and the largest cluster in terms of the total number of data points. Pick them in either order. There may be more clusters of the same size; *T3*: Find and select all the overlapping clusters. The tasks required the users to visually inspect the individual clusters and understand their spatial (*T1* and *T3*) or quantitative (*T2*) relation to other clusters, rather than any detailed knowledge of their specific parameters, such as the exact number of data points in a cluster. As remarked by Wijk [177], such low-level tasks are are often a subject of consideration for information visualization community. The system automatically marked the tasks as completed and displayed a completion message to the participant as soon as all the clusters were selected. Participant were not given information of any errors in their responses.

Table 6.2 The order of the tasks ($T1-T3$) and accompanying datasets ($A-F$) was repeated for each consecutive group of six participants. We chose this ordering as balancing of the datasets was deemed more important as we were primarily interested in the differences between the datasets rather than the particular tasks.

Tasks execution order			**Datasets order**
A(T1,T2,T3)	B(T2,T3,T1)	C(T3,T1,T2)	ABC
A(T2,T3,T1)	C(T3,T1,T2)	B(T1,T2,T3)	ACB
B(T3,T1,T2)	A(T1,T2,T3)	C(T2,T3,T1)	BAC
C(T1,T2,T3)	A(T2,T3,T1)	B(T3,T1,T2)	CAB
B(T2,T3,T1)	C(T3,T1,T2)	A(T1,T2,T3)	BCA
C(T3,T1,T2)	B(T1,T2,T3)	A(T2,T3,T1)	CBA

6.4.4 Results

A General Linear Model three-way mixed repeated measures analysis of variance (ANOVA) was used to analyse task completion times, rotation and travelled distance measurements. Time durations were log-transformed prior to analysis. Error counts were analysed using a Generalized Linear Model using a log-Poisson kernel. All statistical analyses were carried out at an initial significance level of $\alpha = 0.05$, which was adjusted for multiple comparisons with Holm-Bonferroni correction, where applicable.

The NASA TLX, IPQ and SSQ scores were analysed using Friedman's test. However, no significant differences were detected so for brevity we omit these results in the analysis.

Table 6.3 Total counts of the two kinds of errors: "repeated selection" and the "wrong identification" for each task (*T1*, *T2* and *T3*) and dataset (*A*, *B*, *C* and *D*, *E*, *F*) separated by backslash (\).

	A	*B*	*C*	*D*	*E*	*F*
T1	76\76	72\72	55\55	33\33	50\50	31\31
T2	40\85	22\95	13\78	19\66	17\69	7\73
T3	20\53	31\93	35\85	5\73	18\27	64\71

Task Completion Times

The first time-stamp of each task was taken the moment the task scene was loaded. Each participant's selection, including subsequent repeated selections, were also time-stamped and recorded in sequence. The difference between the first time-stamp and the last time-stamp that fulfilled the task's requirements was calculated as task completion time. Fig. 6.3a summarises task completion times.

No statistically significant effects were detected between the two experiments ($F_{1,34} = .487, \eta_p^2 = .012, p = .527$) with respect to task completion times. All the possible interaction combinations were also insignificant. However, there were statistically significant differences between tasks ($F_{2,68} = 36.267, \eta_p^2 = .516, p \leq .001$). Pairwise comparisons of the tasks revealed, as expected, that there were also significant differences ($p \leq .001$) between *T1* and both *T2* and *T3*. The difference between *T2* and *T3* was not significant ($p = .205$).

In other words, the differences between the datasets were insignificant, but the tasks did indeed result in significant differences. This demonstrates that the process of synthesising datasets does result in comparable datasets that can still be used to detect significant differences across tasks.

Errors

For all tasks, an error will occur if the participant repetitively selects any of the previously selected clusters (see Fig. 6.3). In the case of task *T2*, an additional type of error happens if the participant selects neither the smallest, nor the largest, cluster. The same happens if a non-overlapping cluster is selected in task *T3*. The analysis discarded outliers three standard deviations away from the mean.

An analysis using a general linearised model with a log-Poisson kernel revealed that only tasks ($\chi^2 = 9.454, df = 2, p = .009$) are significant predictors of errors whereas the datasets are not. The interaction was insignificant. Linearly independent pairwise comparisons of estimated marginal means revealed a statistically significant difference between *T1* and *T2*

($df = 1, p = .016$). This outcome was to be expected, as tasks *T1* and *T2* required selection of the largest ($N = 15$) and the smallest number ($N = 2$) of clusters respectively (see Tab. 6.1).

The same analysis was repeated for task-specific errors labelled as "incorrect identification" (see Tab. 6.3). Specifically, this included repeated selections in task *T1*, failing to select the largest or the smallest cluster in task *T2*, or selecting a non-overlapping cluster in task *T3*. The differences were not statistically significant for the datasets. However, both tasks ($\chi^2 = 11.678, df = 2, p = .003$) and the interaction ($\chi^2 = 9.588, df = 4, p = .048$) of the factors were statistically significant. Further analysis of linearly independent pairwise comparisons of estimated marginal means revealed statistically significant differences between task *T1* and both task *T2* ($df = 1, p = .002$) and *T3* ($df = 1, p = .018$) respectively. There were also statistically significant differences in the interaction observed between $B(T1)$ & $B(T3)$ ($df = 1, p = .019$), $B(T3)$ & $C(T1)$ ($df = 1, p \leq .001$) and $C(T1)$ & $C(T3)$ ($df = 1, p = .036$).

For the second study, the tasks were significant predictors of errors ($\chi^2 = 14.269, df = 2, p \leq .001$) whereas the datasets were not. An interaction analysis also revealed no statistically significant results. Pairwise comparisons of estimated marginal means revealed statistically significant differences between task *T1* and task *T2* ($df = 1, p = .011$) and *T3* ($df = 1, p = .004$). In addition, pairwise comparisons also revealed statistically significant results between the interaction of dataset/task $D(T1)$ & $D(T3)$ ($df = 1, p = .009$) and $D(T1)$ & $F(T2)$ ($df = 1, p = .037$). As before, only the tasks ($\chi^2 = 8.836, df = 2, p = .012$) were significant predictors of errors, whereas the interaction of tasks and the datasets were not. As in the first study, pairwise comparisons of estimated marginal means revealed a statistically significant difference between task *T1* and task *T2* ($df = 1, p = .008$). Pairwise comparisons of interactions between dataset/task revealed that $D(T3)$ & $E(T3)$ were statistically significant ($df = 1, p = .038$).

Head Rotation and Travelled Distance

To calculate an approximation of the user's head rotation we captured the forward vector extending from the middle of camera's frustum together with changes in the camera position. The angles between successive records of the forward vectors were summed to provide an estimation of total change in the user's head rotation. The analysis of this measurement yielded statistically significant differences between tasks ($F_{2,68} = 34.312, \eta_p^2 = .502, p \leq .001$) and of the interaction between the experiments and the datasets ($F_{2,68} = 3.714, \eta_p^2 = .098, p = .029$), which appears to be driven by datasets B and C (see Fig. 6.3b). No statistically significant effects were observed between the two experiments with respect to head rotation. As expected, pairwise comparisons of the estimated marginal means revealed significant differences between all three tasks, *T1* and *T2* ($p \leq .001$), *T1* and *T3* ($p \leq .001$) and *T2* and *T3* ($p = .004$).

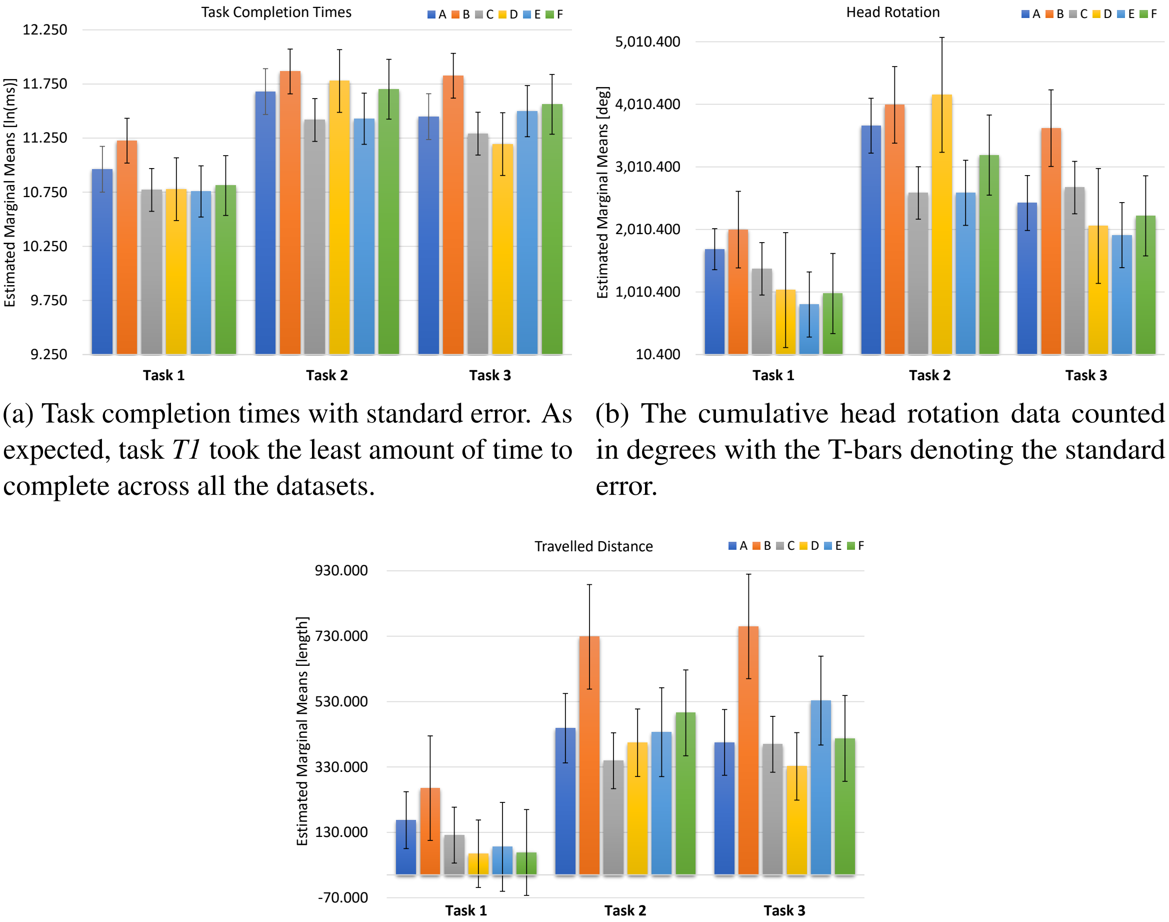
The Euclidean distance between consecutive points were used to compute the length of the user's trajectory in the VR environment; the results are shown in Fig. 6.3c. Statistically significant differences were found between tasks ($F_{2,68} = 19.447, \eta_p^2 = .364, p \leq .001$) and between datasets ($F_{2,68} = 4.857, \eta_p^2 = .125, p = .011$). However, there was no significant difference between the experiments. As expected, a pairwise comparison revealed statistically significant differences between *T1* and *T2* ($p \leq .001$) and *T1* and *T3* ($p \leq .001$). Again, the difference between the datasets appear to be driven by datasets *B* and *C* (see Fig. 6.3c).

We conjecture the differences between the datasets is due to the variation in vertical organisation of the clusters in relation to the natural gaze patterns of the users. In other words, users are more likely to scan the scene laterally than looking up and down. This effect is likely a good indicator of how to place data in the VR environment and to study solutions for mitigating the problem when this is not possible, such as navigation or spatial awareness aids that can assist the user in fully exploiting the VR visualisation.

6.5 Discussion

The validation provides evidence that the generative process for synthesising clustered datasets did indeed result in datasets that can be used in experiments without the datasets themselves being the dominant contributor of the variability between conditions. Overall, the tasks revealed significant differences, which was to be expected, while the influences of the datasets as an independent variable in the experimental design did not result in significant differences, even though we used statistical significance tests with as high statistical power as possible and a weak significance level of $\alpha = 0.05$. Furthermore, effect sizes have been reported throughout, which help quantify the proportion of the explained variability. This supports the hypothesis that the generative process described in this Chapter is suitable for iterative A/B testing-based design and experimentation.

The generative process can greatly simplify such testing by eliminating the learning effect of using a single dataset by instead introducing participants to series of generated synthetic datasets that are unlikely to be dominant causal variables explaining differences in performance across conditions. The differences observed between the tasks were to be expected, specifically as tasks *T2* and *T3* required a deeper understanding of the relation between the individual clusters, especially compared to task *T1*. Further, all of the tasks required participants to gain some understanding of the spatial distribution of the clusters, which, as



(a) Task completion times with standard error. As expected, task *T1* took the least amount of time to complete across all the datasets.

(b) The cumulative head rotation data counted in degrees with the T-bars denoting the standard error.

(c) Total travelled distance in Unity3D's native units with the T-bars denoting the standard error. As expected, the least amount of movement was observed in the task *T1* across all the datasets.

the participant is fully immersed in the data and retains only a first-person perspective, can only become apparent to the user over time.

Finally, another factor that may impact the overall differences in task completion times is the user's selection strategy. The most obvious tactic, which would yield the fastest execution times while simultaneously causing, on average, the most errors, would be to select all clusters as quickly as possible under any and all experimental conditions. However, the recorded data indicates that no participant used this approach.

6.6 Conclusion

This Chapter has presented a new approach that simplifies iterative A/B testing of cluster-based VR analytics tasks. By robustly generating synthetic clustered datasets that give rise to similar user behaviour but are still perceived as different by participants, it is possible to A/B test successive interactions of interaction techniques and VR analytics tools without being overly concerned of the learning effect of using a single natural dataset, or the heterogeneity induced by using several natural datasets, whose underpinning properties are uncontrollable. The process can be further enhanced by tailoring the algorithms to facilitate individual needs.

The prior work described in Chapter 2 has proposed and tested other forms of synthetic data generation [78, 9, 184]. However, none of these papers actually check whether the dataset is itself an explanatory variable. This introduces a risk of a systematic methodological error in the literature as failing to control for the dataset reduces internal validity. To ensure rigorous findings from controlled experiments it is vital to validate the instrument, in this case, the dataset. This requires developing a specific replicable and transparent mechanism for generating synthetic datasets *and* carrying out specific controlled experiments that ensure the dataset is indeed not an explanatory variable. Moreover, some of the datasets features described in Chapter 2 can be easily incorporated to aid a generative process as the one described here. For example, outliers can be randomly added to a particular cluster or to the dataset as a whole by randomly generating data points outside of clusters or their bounding volumes.

The generative method has the potential to be generalised to synthesise multidimensional data. This involves replacing the 3D trail with an n-dimensional Brownian motion trail to determine the clusters' placements within the n-dimensional space. As a consequence, the colour-coding scheme is then no longer viable, as the data is now unsuitable for visualisation as a 3D point cloud. One alternative is parallel coordinates (see [59, 151]). Further, a high-dimensional clustering algorithm [183] needs to be applied to determine the range of overlaps between the individual clusters.

In summary, this Chapter has demonstrated that artificially generating datasets is a viable method. However, the caution is of the essence when implementing this method as part of an empirical study. Similar to how repeated measures designs should always be checked for asymmetrical skill-transfer effects [108], it is recommend to analyse the dataset as an independent variable to reaffirm that the dataset was not a significant contributor in a particular study.

Finally, while the frequentist statistics was used when analysing the results, the generative method addresses a general experimental design problem and the solution is therefore applicable to other types of statistical inference, such as Bayesian inference.

Chapter 7

Conclusions

7.1 Summary of Contributions

The contributions of this research concerning the exploration of engineering applications of immersive analytics in *virtual reality* (VR) focused on a number of main research questions (**RQ1—RQ5**) outlined in Chapter 1. These five research questions can be further subdived into a range of smaller, more centred queries. The following subsections will briefly summarise the key contributions presented in each of the previous chapters grouped by the scope of the main research question that they were addressing. Results of the conducted user experiments and analysis of recorded data as well as the observations of the users' behaviour allowed to trim the vast and unexplored design space of VR-based immersive analytics systems.

7.1.1 Selection of Content Source for Virtual Reality

RQ1.1 What practical methods are available for providing content for virtual reality?

While rapid advances in VR technologies have shown promise for consumer applications, usage in professional settings remains limited. Professional use, such as engineering applications, requires task-specific models reducing the plasticity and reusability of preexisting content. The generation of suitable models remains a crucial area of exploration to support professional adoption of VR. Section 1.4 in Chapter 1 briefly describes three techniques for generating 3D models: (A) computer-aided design (CAD), a designer-intensive method using popular software modelling packages; (B) photogrammetry, a technique to assemble models from photographic and sensor data of real-world objects; and (C) generative processing, a nascent, yet promising, technique for generating numerous varied and distinctive models using a machine learning-based approach.

In addition, the feasibility of using two of these methods, were tested and discussed in another two chapters. For instance, (A) the CAD models were used in both use cases described in Chapter 3. Whereas, Chapter 4 evaluates the feasibility of using (B) photogrammetric 3D models to conduct offline engineering surveying. These examples were partially answering the question of these two methods being a viable solution for providing quality-content for VR as investigated for **RQ1.1**.

RQ1.2 What is a meaningful way of comparing content preparation methods?

Appropriate fidelity of the 3D models for the VR environments, while dependant on their intended use, is pivotal in supporting the highly sensitive applications common in professional use. Such uses include modelling, review, experimentation, analysis, and archiving. To this end, Section 1.4 of Chapter 1.4 compares and contrasts three techniques for generating 3D models: (A) computer-aided design (CAD), (B) photogrammetry, and (C) generative approach. To conduct the comparison exercises, Chapter 1.4 provided and discussed six carefully selected characteristics that offer a framework for review of these three methods. This exercise allowed us to distil and talk about the advantages and disadvantages of each approach to providing quality content for VR. Moreover, using three usage scenarios, the weighting system associated with each category has been provided. These weights are in a way describing importance which may be different for different stakeholders and even use cases but can aid designers and developers in deciding on a suitable technique. The appropriate synthesis of these methods is important for the system designers, as the methods themselves, to some extent, both determine and extend the potential application areas and particular use cases within them.

7.1.2 Applications in Aerospace and Aeronautics

RQ2.1 Can virtual reality analytics be used as a feasible alternative for design engineering in aerospace and aeronautics?

Chapter 3 introduced the *AeroVR*—an immersive aerospace design environment with the objective of aiding the aerodynamic component design process by interactively visualising performance and geometry. In here, I decomposed the design of such an environment into function structures, identify the primary and secondary tasks, present an implementation of the system, and verify the interface in terms of usability and expressiveness. The AeroVR was also deployed on a prototypical design study of a compressor blade for an engine. Here, I had also explored the feasibility of using gaze-tracking coupled with the use of a hand-held gamepad controller to interact with such an immersive system for design engineering in the aerospace and aeronautics domains. However, the heuristic assessment of the usability and effectiveness of this interface should be further reflected up after carrying out and analysing results of user experiments.

RQ2.2 Can virtual reality analytics be used as a feasible tool for digital twinning in aerospace and aeronautics?

Chapter 3 has shown an immersive environment where VR is used to visualise the performance of a fleet of aircraft engines. This virtual environment uses 3D geometric computer-aided design (CAD) models of the engines paired with performance maps that characterise their nominal working condition. These maps plot pressure ratio and efficiency as a function of shaft speed and inlet flow capacity for the numerous engine sub-systems. Superimposed on these maps is the true performance of each engine, obtained through real-time sensors. In this bespoke virtual space, an engineer can rapidly analyse the health of different engine sub-systems across the fleet within seconds. One of the key elements of such a system is the selection of an appropriate interaction technique.

Chapter 3 also reports on an observational study with a small number of domain-experts to identify usability problems, spot potential improvements, and gain insights into our design interaction capabilities. This study allowed us to trim the design space and to guide further design efforts in this area by analysing the qualitative feedback provided by the end-users. Aerospace systems are rarely used outside of specialised industrial environments, and typically they are operated by highly qualified personnel. However, as all the participants were domain-experts, the results of this study may not generalise to a wider population. Due to this, and the challenges of recruiting large numbers of domain experts to support statistical analysis, additional more general user studies should be designed to consider a broader context such as proxemics or hand-tracking and gesture recognition for model manipulation in VR to further validate these findings.

7.1.3 Applications in Civil Engineering

RQ3.1 What are the design requirements for a virtual reality analytics system for off-line engineering surveying?

Recent advancements in VR may help to unlock the full potential offered by 3D photo-realistic models generated using state-of-art photogrammetric methods. Such models are becoming a viable alternative for providing high-quality content for immersive systems as well as become a basis for digital twinning. Using VR to carry out analyses on photogrammetric models has the potential to improve and assist the user in performing basic off-line engineering inspection of a digitised object.

Chapter 4 presents an immersive, gesture-controlled system for manipulation and inspection of 3D photogrammetric models of physical objects in VR. An observational study with a group of domain-expert participants validated the feasibility of the system design. This system was populated with a 3D photogrammetric models of an existing building and the domain experts were asked to carry out a survey measurement of the object using the offered measurement toolbox. This allowed us to distil and reason about the design requirements presented in-detail in Chapter 4. Moreover, this system shows that VR twins can be used as a viable substitute for some civil engineering purposes. Such systems may have applications allowing additional surveys of structures for increased safety, while potentially reducing both risk to surveyors and cost to businesses and regulators.

RQ3.2 Is it viable to use gestural input to carry out off-line engineering surveying in virtual reality?

The study in Chapter 4 confirmed the potential of the VR-based systems to be applied in practical real-word cases of off-line inspections of buildings and other 3D structures. Moreover, the system provided the users with a bimanual-interaction technique coupled with the head-based gaze-tracking capabilities. The manipulation of the object occurs via the user's gestural input, i.e. the hand-tracking facilitated with a depth sensor. The gestural input coupled with the use of gaze-tracking for selecting objects which user wants to interact with, was the basis of the main interaction method incorporated in the system.

To this end, Chapter 4 also reports the results of an observational study with a small group of domain-experts. This study explored the possibility of using the hand-tracking and gesture-recognition capabilities coupled with the gaze-tracking within an immersive, VR environment. The qualitative data and the observations of the participants' behaviour and their comments allowed to reason about the feasibility of using such interaction technique to extract relevant data from 3D photogrammetric models whilst being immersed in VR. The results of the building model's off-line inspection carried out by the participants suggest that the experts were able to take the measurements within a reasonable level of accuracy. Moreover, all the participants were able to promptly gain fluency in manipulating the model with their hand gestures, as was confirmed by their own comments. However, as the experiment had the form of a qualitative observational study, there is a possibility that these results may hint at additional confounding effects. Notably, there was an observed difference in the behaviour of the least work experienced and simultaneously the youngest participant as opposed to other users concerning the model rescaling. This observation could be further investigated in the randomised quantitative study to confirm or reject potential causes of such behaviour.

7.1.4 Applications in Computational Design

RQ4.1 What are the design requirements for immersive parallel coordinates plots in computational design engineering?

Computational engineering design methods and tools are common practice in modern industry. Such approaches are integral in enabling designers to efficiently explore larger and more complex design spaces. However, at the same time, computational engineering design methods tend to dramatically increase the number of candidate solutions that decision-makers must interpret in order to make appropriate choices within a set of solutions. The goal of this work was to enhance the design decision-making process by embedding visual analytics into an immersive virtual reality environment. To this end, Chapter 5 presented a system called IPCP: *Immersive Parallel Coordinates Plots*. IPCP combines the well-established parallel coordinates visualisation technique for high-dimensional data with immersive virtual reality. The same chapter also presents a number of design requirements that have to be fulfilled in order for the system provide the user with a viable alternative for 2D environments. The design of this system relying on the captured requirements allows the representation and exploration of multidimensional scientific datasets.

RQ4.2 Can immersive parallel coordinates plots assist the user in discovering new knowledge from an abstract dataset in computational design engineering?

Since all candidate solutions can be represented in digital form together with their assessment criteria, evaluated according to some sort of simulation model, a natural way to explore and understand the complexities of the design problem is to visualise their multidimensional nature. The task now involves the discovery of patterns and trends within the multidimensional design space.

To this end, Chapter 5 proposed the usage of IPCP to exploit and discover efficient means to use new technology within a conventional decision-making process. The aim is to provide benefits by enhancing visualisations of 3D geometry and other physical quantities with scientific information. A qualitative evaluation with two surrogate expert users, knowledgeable in multidimensional data analysis, demonstrated that the system can be used successfully to detect both known and previously unknown patterns in a real-world test dataset, producing an early indicative validation of its suitability for decision support in engineering design processes. However, this sample size is very small. Hence, to draw any more tangible conclusions, controlled user studies with a substantially larger number of participants should be carried out to allow these findings to be further validated and properly contextualised before any generalisation to a broader user base .

7.1.5 Evaluation with Artificial Datasets

RQ5 Can artificially synthesised datasets be used to meaningfully evaluate virtual reality analytics interfaces?

Virtual reality (VR) is a promising technology platform for immersive visual analytics. However, the design space of VR analytics interface design is vast and difficult to explore using traditional A/B comparisons in controlled experiments. One factor that complicates such comparisons is the used dataset. Exposing participants to the same dataset in all conditions introduces a learning effect. On the other hand, using different datasets for all experimental conditions introduces the dataset as an uncontrolled variable. Chapter 6 proposes to rectify this by introducing a generative process for synthesising clustered datasets for VR analytics experiments. This process generates datasets that are distinct while simultaneously allowing systematic comparisons in experiments. In addition, these datasets can also be used in an iterative design process. In a two-part experiment, we demonstrated the validity of the process and shown how new insights in VR-based analytics can be gained using synthetic datasets.

7.2 Limitations and Future Work

This section enlists a number of potential avenues for future work with respect to the research outlined in this thesis. It also contains remarks with regards to some limitations of the performed studies as well as how they can be addressed in future research.

The general concerns are related to the confounding bias often occurring in observational studies as the ones described in Chapters 3–5. Other factors that may cause bias in the data include the size of data samples, which prevent an appropriate statistical analysis, and the participants' individual biases. The latter may include their prior experience or lack thereof with respect to VR or particular interaction techniques. This novelty, and direct engagement of the participants with the researcher, may have encouraged or caused participants to feel obliged to judge the interface positively. Moreover, as all the qualitative experiments relied on domain expert knowledge, the findings may not easily generalise to non-expert users. In this context, where appropriate, I will briefly discuss how these challenges can be overcome in future work.

7.2.1 Virtual Reality Analytics for Aerospace and Aeronautics

Aerospace Design

The developed system supports the complete 3D design of a turbomachinery component in VR. In addition to the sufficient summary plots, the goal here was to incorporate characteristics of blade performance at multiple operating points and have reduced order models to estimate the performance characteristics of new designs.

To trim the vast design space, this system was heuristically verified in terms of usability [102, 101] and effectiveness [45, 46] using formative evaluation methods. However, to provide a more tangible evaluation, this system has to be further studied in a series of controlled, qualitative user experiments. The results of these studies can help to assess the advantages and disadvantages of using particular interaction techniques or allowing sensible comparison of the VR-based interface with a standard input/output devices such as computer screen, mice, and keyboard.

In forthcoming years, the cost of the headset, and the required computing resources, will be further minimised with the introduction of next-generation of controllers, wireless headsets, and gestures, or speech-based interfaces. Simultaneously, these rapid advancements in hardware and software open up completely new possibilities in terms of interaction techniques, rapid information analysis and the amount of data that can be processed and visualised at once. The two most promising venues of further development of the *AeroVR* system are the investigation of which interaction techniques may bring the most benefit to the user and integration of our system with a system operating on knowledge from a domain expert, i.e. a *knowledge-based system* [66, 100]. The former would include adding either controller-based laser-pointing or hand-tracking capabilities or a combination of thereof depending on the particular user's needs. The latter would require to develop and integrate a *knowledge-(data)base* [66] with the interface provided by our system to build a *knowledge-based system* [66, 100].

Digital Twinning

There is a number of potential avenues exposed by the conducted work for future research. One of them concerns 2D graphs. In Chapter 3, P3 commented that representing a 2D graph in 3D space had no significant advantages over that in 2D space. He suggested allowing multiple dimensions which could make better use of the VR environment. As discussed in

DI3: Complex Graphs, we could, for instance, extend the iso-contour lines into the Z-plane in the plot's local coordinate system to form a 3D-like structure. This maybe particularly useful as outlier points, such as the one the participants had to find in Task 3(b), could more easily be spotted as they would be on top of such plot if rotated along the X-axis. In addition, such plots could be easily rotated using the bimanual manipulation method.

It may also be interesting to investigate the different alignment of the aeroengine models. For instance, as proposed by one of the study participants, we could include the functionality similar to one offered by a conveyor belt. In this case we could, for example, place the engines on a conveyor belt, or rotate the engines on an arc using the *swipe gesture*.

These potential research directions represent future avenues for research exposed by this work. These investigations would broaden the presented techniques and further develop promising gestural-based input method as means of interaction for 3D immersive analytics. Moreover, the carefully designed randomised trials with a number of participants warranting an appropriate statistical analysis can be used to confirm or reject the preliminary findings of this work. Similar to the previous study, the users may be tasked with identifying the faulty subsystem of an aeroengine using the VR environment as well as the standard computer setup. Here, the system may record various qualitative data such as the task execution times for comparison between the two systems.

7.2.2 Virtual Reality Analytics for Civil Engineering

One challenge exposed in Chapter 4 was the snapping grid technique. Future work could revise the snapping grid functionality by, for example, keeping the snapping range constant and fairly small even when the user enlarges or decreases the size of the model. Moreover, refinements to the algorithm for snapping grid generation may be more efficient and effective in terms of placing the snapping points by automatically choosing points of interest, such as mesh vertices. These tweaks could be tested in a series of controlled experiments to assess if, and by how much, the improved snapping grid is useful and effective. Further, the system could be extended with the capability of handling multiple models at once. In this scenario, the models could be manipulated by the user as either a single object or a group thereof. The controlled experiments with a substantially large sample size may, to a large extent, prevent the existence of some of the confounding factors. For instance, six of the participants (P1-P6) reported limited or no prior experience with the VR, and only a single participant did not have an engineering background (P1). However, it was observed in the experiment that the person who behaved notably differently with regards to model manipulation was the youngest and least experienced participant (P2) and not the participant without proper engineering training (P1). To further investigate this observation, a controlled, properly randomised study has to be carried out to allow further interpretation of the identified outlier results and potentially discover any number of causal factors that may have affected the user's behaviour, such as P2's age or work experience.

7.2.3 Virtual Reality Analytics for Computational Design

A promising area of future work is to further refine the presented visualisation system by evolving the task model and the underpinning functions and revisit the process of translating functions to solutions. This can be achieved by identifying controllable and uncontrollable parameters for each overall function necessary to carry out a task and study the effects of modifying these parameters in isolation. The current IPCP version can aid future explorations of how to make informed decisions on, for instance, which combination of inputs and outputs to choose to arrive at an effective operating point for immersive data analysis and thereby assist in gaining a more complete understanding of the trade-offs and their effects that are typically inherent in such design decisions. Such technological capability will be inherently harmonised with the state-of-the-art interactive computational engineering design methodologies [56] opening new avenues of research for the next generation engineering design software environment and systems.

7.3 Final Remarks

Among different types of so-called immersive interfaces, *virtual reality* (VR) carries a lot of potential as it is a relatively mature technology compared to optical see-through head-mounted displays. VR is a term interchangeably used to denote both various headsets and fully immersive 3D environments that these devices transport their users to. When working with such interfaces, the VR designer can leverage a number of different visualisation and interaction techniques to design the most effective and compelling user experiences for specific use-cases. In turn, the combination of interactive visualisation with analytical reasoning about the visualised data bestowed by the VR environment forms a foundation for immersive analytics applications. However, due to its relative novelty, the full potential of VR for immersive analytics in engineering applications has to be yet fully understood. The first step towards this goal is a careful consideration of where such technology can bring the most benefits to the users.

As demonstrated in this thesis, VR-based analytics has a wealth of potential application areas in engineering. One of the most promising application domains of such immersive interfaces in the industry is various branches of engineering, including aerospace and aeronautics, and civil engineering. The range of potential applications is vast and increasing as new industries are adopting immersive tools. However, the use of immersive technologies brings its own challenges. For example, the type of content used to populate the VR environment will simultaneously determine and extend the particular application domain.

To this end, the work presented in this thesis has focused on designing and analysing various interactive VR-based immersive systems for immersive visual analytics in engineering. For example, I have designed, developed, and tested systems for digital twin assessment and design engineering in aerospace, as well as a system for extracting information from photo-realistic models of existing structures.

As VR continues to mature, I believe it has large potential across a wide variety of application domains. To support future exploration, this thesis offers broadly applicable overviews of both the underlying technologies required to support VR and detailed analysis on approaches and techniques that can be used to populate such environments. Specifically, however, this work serves both as a series of case studies demonstrating the viability of VR technologies in engineering, but also highlights the potential approaches and pitfalls to applying VR in new domains.

References

[1] Aber, J. S., Marzolff, I., and Ries, J. B. (2010). Chapter 3 - photogrammetry. In Aber, J. S., Marzolff, I., and Ries, J. B., editors, *Small-Format Aerial Photography*, pages 23 – 39. Elsevier, Amsterdam.

[2] Absil, P.-A., Mahony, R., and Sepulchre, R. (2009). *Optimization algorithms on matrix manifolds*. Princeton University Press, 41 William St. Princeton, NJ, United States.

[3] Achlioptas, P., Diamanti, O., Mitliagkas, I., and Guibas, L. (2017). Learning representations and generative models for 3d point clouds. *arXiv preprint arXiv:1707.02392*.

[4] American Society for Photogrammetry and Remote Sensing (Last accessed: Nov. 2019). What is ASPRS? https://www.asprs.org/organization/what-is-asprs.html.

[5] Ankerst, M., Breunig, M. M., Kriegel, H.-P., and Sander, J. (1999). Optics: Ordering points to identify the clustering structure. *SIGMOD Rec.*, 28(2):49–60.

[6] Antlej, K., Bykersma, M., Mortimer, M., Vickers-Rich, P., Rich, T., and Horan, B. (2018). Real-world Data for Virtual Reality Experiences: Interpreting Excavations. In *2018 3rd Digital Heritage International Congress (DigitalHERITAGE) held jointly with 2018 24th International Conference on Virtual Systems Multimedia (VSMM 2018)*, pages 1–8.

[7] Artero, A. O., Oliveira, M. C. F. d., and Levkowitz, H. (2004). Uncovering Clusters in Crowded Parallel Coordinates Visualizations. In *IEEE Symposium on Information Visualization*, pages 81–88.

[8] Azai, T., Otsuki, M., Shibata, F., and Kimura, A. (2018). Open Palm Menu: A Virtual Menu Placed in Front of the Palm. In *Proceedings of the 9th Augmented Human International Conference*, AH '18, pages 17:1–17:5, New York, NY, USA. ACM. event-place: Seoul, Republic of Korea.

[9] Bach, B., Sicat, R., Beyer, J., Cordeil, M., and Pfister, H. (2018). The hologram in my hand: How effective is interactive exploration of 3d visualizations in immersive tangible augmented reality? *IEEE Trans. Vis. Comput. Graphics*, 24:457–467.

[10] Bach, B., Spritzer, A., Lutton, E., and Fekete, J.-D. (2012). Interactive Random Graph Generation with Evolutionary Algorithms. In Didimo, W. and Patrignani, M., editors, *Graph Drawing 2012*, volume 7704 of *Lecture Notes in Computer Science*, pages 541–552, Berlin, Germany. Springer.

[11] Balakrishnan, R. and Kurtenbach, G. (1999). Exploring bimanual camera control and object manipulation in 3D graphics interfaces. In *Proceedings of the SIGCHI conference on Human Factors in Computing Systems*, CHI '99, pages 56–62, Pittsburgh, Pennsylvania, USA. Association for Computing Machinery.

[12] Bouchard, S., Robillard, G., and Renaud, P. (2007). Revising the factor structure of the Simulator Sickness Questionnaire. *Annual Review of CyberTherapy and Telemedicine (ARCTT)*, 5:117–122.

[13] Brath, R. (2014). 3D InfoVis is here to stay: Deal with it. In *2014 IEEE VIS International Workshop on 3DVis (3DVis)*, pages 25–31.

[14] Butler, L., Lau, D., Gregory, A., Girolami, M., and Elshafie, M. (2019). Introducing data-centric engineering to instrumented infrastructure. In *International Conference on Smart Infrastructure and Construction 2019 (ICSIC) Driving data-informed decision-making*, pages 343–349. ICE Publishing.

[15] Butscher, S., Hubenschmid, S., Müller, J., Fuchs, J., and Reiterer, H. (2018). Clusters, Trends, and Outliers: How Immersive Technologies Can Facilitate the Collaborative Analysis of Multidimensional Data. In *Proceedings of the 2018 CHI Conference on Human Factors in Computing Systems*, CHI '18, pages 90:1–90:12, New York, NY, USA. ACM.

[16] Buxton, W. and Myers, B. (1986). A study in two-handed input. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '86, pages 321–326, Boston, Massachusetts, USA. Association for Computing Machinery.

[17] Caputo, F. M. and Giachetti, A. (2015). Evaluation of Basic Object Manipulation Modes for Low-cost Immersive Virtual Reality. In *Proceedings of the 11th Biannual Conference on Italian SIGCHI Chapter*, CHItaly 2015, pages 74–77, New York, NY, USA. ACM. event-place: Rome, Italy.

[18] Carroll, J. M. and Carrithers, C. (1984). Training wheels in a user interface. *Communications of the ACM*, 27(8):800–806.

[19] Chandler, T., Cordeil, M., Czauderna, T., Dwyer, T., Glowacki, J., Goncu, C., Klapperstueck, M., Klein, K., Marriott, K., Schreiber, F., and Wilson, E. (2015). Immersive analytics. In Engelke, U., Heinrich, J., Bednarz, T., Klein, K., and Nguyen, Q., editors, *2015 Big Data Visual Analytics (BDVA)*, United States. IEEE, Institute of Electrical and Electronics Engineers.

[20] Chang, A. X., Funkhouser, T., Guibas, L., Hanrahan, P., Huang, Q., Li, Z., Savarese, S., Savva, M., Song, S., Su, H., et al. (2015). Shapenet: An information-rich 3d model repository. *arXiv preprint arXiv:1512.03012*.

[21] Chang, C., Dwyer, T., and Marriott, K. (2018). An Evaluation of Perceptually Complementary Views for Multivariate Data. In *2018 IEEE Pacific Visualization Symposium (PacificVis)*, pages 195–204.

[22] Clark, A. and Moodley, D. (2016). A System for a Hand Gesture-Manipulated Virtual Reality Environment. In *Proceedings of the Annual Conference of the South African Institute of Computer Scientists and Information Technologists*, SAICSIT '16, pages 10:1–10:10, New York, NY, USA. ACM. event-place: Johannesburg, South Africa.

[23] Clergeaud, D., Guillaume, F., and Guitton, P. (2016). 3D collaborative interaction for aerospace industry. In *2016 IEEE 3rd VR International Workshop on Collaborative Virtual Environments (3DCVE)*, pages 13–15.

[24] Cohen, R. D. (1986). Self similarity in Brownian motion and other ergodic phenomena. *J. Chem. Educ.*, 63(11):933.

[25] Constantine, P. G. (2015). *Active Subspaces: Emerging Ideas for Dimension Reduction in Parameter Studies*, volume 2. SIAM Spotlights, 3600 Market Street, 6th Floor, Philadelphia, PA 19104-2688 USA.

[26] Constantine, P. G., Eftekhari, A., Hokanson, J., and Ward, R. A. (2017). A near-stationary subspace for ridge approximation. *Computer Methods in Applied Mechanics and Engineering*, 326:402–421.

[27] Cook, R. D. (2009). *Regression graphics: Ideas for studying regressions through graphics*, volume 482. John Wiley & Sons, Inc, Hoboken, New Jersey.

[28] Cordeil, M., Cunningham, A., Dwyer, T., Thomas, B. H., and Marriott, K. (2017). ImAxes: Immersive Axes As Embodied Affordances for Interactive Multivariate Data Visualisation. In *Proceedings of the 30th Annual ACM Symposium on User Interface Software and Technology*, UIST '17, pages 71–83, New York, NY, USA. ACM.

[29] Csikszentmihalyi, M. (2000). *Beyond boredom and anxiety*. Beyond boredom and anxiety. Jossey-Bass, San Francisco, CA, US. Pages: xxx, 231.

[30] Cui, J., Kuijper, A., Fellner, D. W., and Sourin, A. (2016). Understanding People's Mental Models of Mid-Air Interaction for Virtual Assembly and Shape Modeling. In *Proceedings of the 29th International Conference on Computer Animation and Social Agents*, CASA '16, pages 139–146, New York, NY, USA. ACM. event-place: Geneva, Switzerland.

[31] Cynthia, B., Harrower, M., Sheesley, B., Woodruff, A., and Heyman, D. ([Online]: Nov. 2018). Colorbrewer2.0. http://colorbrewer2.org/.

[32] D'Ambros, A., Kipouros, T., Zachos, P., Savill, M., and Benini, E. (2018). Computational Design Optimization for S-Ducts. *Designs*, 2(36):1–21.

[33] Dang, T. N., Wilkinson, L., and Anand, A. (2010). Stacking Graphic Elements to Avoid Over-Plotting. *IEEE Transactions on Visualization and Computer Graphics*, 16(6):1044–1052.

[34] Einstein, A. (1905). Über die von der molekularkinetischen Theorie der Wärme geforderte Bewegung von in ruhenden Flüssigkeiten suspendierten Teilchen. *Annalen der Physik*, 322(8):549–560.

[35] El Saddik, A. (2018). Digital Twins: The Convergence of Multimedia Technologies. *IEEE MultiMedia*, 25(2):87–92. Conference Name: IEEE MultiMedia.

[36] Ester, M., Kriegel, H.-P., Sander, J., and Xu, X. (1996). A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining*, KDD'96, pages 226–231. AAAI Press.

[37] Falkman, G. (2001). Information visualisation in clinical Odontology: multidimensional analysis and interactive data exploration. *Artificial Intelligence in Medicine*, 22(2):133–158.

[38] Fanea, E., Carpendale, S., and Isenberg, T. (2005). An interactive 3d integration of parallel coordinates and star glyphs. In *IEEE Symposium on Information Visualization, 2005. INFOVIS 2005.*, pages 149–156.

[39] Filho, J. A. W., Rey, M. F., Freitas, C. S., and Nedel, L. (2018). Immersive visualization of abstract information: An evaluation on dimensionally-reduced data scatterplots. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 483–490, Los Alamitos, CA, USA. IEEE Computer Society.

[40] Fritsch, D. and Klein, M. (2017). 3D and 4D Modeling for AR and VR App Developments. In *2017 23rd International Conference on Virtual System Multimedia (VSMM)*, pages 1–8.

[41] G. Farin, J. Hoschek, M. K. (2002). *Handbook of Computer Aided Geometric Design*, volume 130. Elsevier Science, 1st edition.

[42] García-Hernández, R. J., Anthes, C., Wiedemann, M., and Kranzlmüller, D. (2016). Perspectives for using virtual reality to extend visual data mining in information visualization. In *2016 IEEE Aerospace Conference*, pages 1–11.

[43] Glaessgen, E. and Stargel, D. (2012). The digital twin paradigm for future nasa and us air force vehicles. In *53rd AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference 20th AIAA/ASME/AHS Adaptive Structures Conference 14th AIAA*, page 1818.

[44] Gonzaga, L., Roberto Veronez, M., Kannenberg, G. L., Alves, D. N., Cazarin, C. L., Santana, L. G., Fraga, J. L. d., Inocencio, L. C., Souza, L. V. d., Marson, F., Bordin, F., and Tognoli, F. M. W. (2018). Immersive Virtual Fieldwork: Advances for the Petroleum Industry. In *2018 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 561–562.

[45] Green, T. R. G. (1989). Cognitive dimensions of notations. In *People and Computers V*, pages 443–460. University Press.

[46] Green, T. R. G. (1990). The Cognitive Dimension of Viscosity: A Sticky Problem for HCI. In *Proceedings of the IFIP TC13 Third Interational Conference on Human-Computer Interaction*, INTERACT '90, pages 79–86, Amsterdam, The Netherlands, The Netherlands. North-Holland Publishing Co.

[47] Greyman studios S.L. (Last accessed: July 2017). Off Screen Indicator. https://assetstore.unity.com/packages/tools/gui/off-screen-indicator-57062.

[48] Gröller, E., Löffelmann, H., and Wegenkittl, R. (1997). Visualization of Analytically Defined Dynamical Systems. In *Scientific Visualization Conference (dagstuhl '97)*, pages 71–71.

[49] Grossman, T. and Balakrishnan, R. (2005). The Bubble Cursor: Enhancing Target Acquisition by Dynamic Resizing of the Cursor's Activation Area. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '05, pages 281–290. ACM. event-place: Portland, Oregon, USA.

[50] Guiard, Y. (1987). Asymmetric Division of Labor in Human Skilled Bimanual Action. *Journal of Motor Behavior*, 19(4):486–517. Publisher: Routledge _eprint: https://doi.org/10.1080/00222895.1987.10735426.

[51] Ha, D., Dai, A., and Le, Q. V. (2016). Hypernetworks. *arXiv preprint arXiv:1609.09106*.

[52] Hale, J. P. (1994). Applied virtual reality in aerospace design. In *Proceedings of WESCON '94*, pages 378–383.

[53] Hart, S. G. and Staveland, L. E. (1988). Development of NASA-TLX (Task Load Index): Results of Empirical and Theoretical Research. In *Advances in Psychology*, volume 52, pages 139–183. Elsevier.

[54] Havlin, S. and Ben-Avraham, D. (1987). Diffusion in disordered media. *Advances in Physics*, 36(6):695–798.

[55] Heinrich, J. and Weiskopf, D. (2015). Parallel Coordinates for Multidimensional Data Visualization: Basic Concepts. *Computing in Science Engineering*, 17(3):70–76.

[56] Hettenhausen, J., Lewis, A., Randall, M., and Kipouros, T. (2013). Interactive multi-objective particle swarm optimisation using decision space interaction. In *2013 IEEE Congress on Evolutionary Computation*, pages 3411–3418.

[57] Heyns, M., https://grabcad.com/michael.heyns (Last accessed: September 2019). Trent 1000 high bypass turbofan, https://grabcad.com/library/trent-1000-high-bypass-turbofan. GrabCAD https://grabcad.com.

[58] Hokanson, J. M. and Constantine, P. G. (2018). Data-driven polynomial ridge approximation using variable projection. *SIAM Journal on Scientific Computing*, 40(3):A1566–A1589.

[59] Holten, D. and Wijk, J. J. V. (2010). Evaluation of Cluster Identification Performance for Different PCP Variants. *Computer Graphics Forum*, 29(3):793–802.

[60] Hunter, J. D. (2007). Matplotlib: A 2D graphics environment. *Computing In Science & Engineering*, 9(3):90–95.

[61] Igroup ([Online]: Nov. 2018). Survey on experiences in virtual worlds. http://www.igroup.org.

[62] Inselberg, A. (1985). The plane with parallel coordinates. *The Visual Computer*, 1(2):69–91.

[63] Inselberg, A. (2009). *Parallel Coordinates: Visual Multidimensional Geometry and Its Applications*. Springer, New York, USA, 1 edition edition.

[64] Isenberg, T., Isenberg, P., Chen, J., Sedlmair, M., and Möller, T. (2013). A Systematic Review on the Practice of Evaluating Visualization. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2818–2827.

[65] Ishihara, S. (2017). *Ishihara's Tests for Colour Deficiency*. Kanehara Trading Inc, Tokyo, Japan, 38 plates edition edition.

[66] Jarke, M., Neumann, B., Vassiliou, Y., and Wahlster, W. (1989). KBMS Requirements of Knowledge-Based Systems. In Brodie, M. L., Mylopoulos, J., Schmidt, J. W., Schmidt, J. W., and Thanos, C., editors, *Foundations of Knowledge Base Management*, pages 381–394. Springer Berlin Heidelberg, Berlin, Heidelberg.

[67] Jeong, S., Chiba, K., and Obayashi, S. (2005). Data Mining for Aerodynamic Design Space. In *23rd AIAA Applied Aerodynamics Conference*. American Institute of Aeronautics and Astronautics.

[68] Johansson, J., Forsell, C., and Cooper, M. (2014). On the usability of three-dimensional display in parallel coordinates: Evaluating the efficiency of identifying two-dimensional relationships. *Information Visualization*, 13(1):29–41.

[69] Johansson, J., Ljung, P., Jern, M., and Cooper, M. (2006). Revealing Structure in Visualizations of Dense 2d and 3d Parallel Coordinates. *Information Visualization*, 5(2):125–136.

[70] Juckette, C., Richards-Rissetto, H., Aldana, H. E. G., and Martinez, N. (2018). Using Virtual Reality and Photogrammetry to Enrich 3d Object Identity. In *2018 3rd Digital Heritage International Congress (DigitalHERITAGE) held jointly with 2018 24th International Conference on Virtual Systems Multimedia (VSMM 2018)*, pages 1–5.

[71] Jude, A., Poor, G. M., and Guinness, D. (2016). Grasp, Grab or Pinch? Identifying User Preference for In-Air Gestural Manipulation. In *Proceedings of the 2016 Symposium on Spatial User Interaction*, SUI '16, pages 219–219, New York, NY, USA. ACM. event-place: Tokyo, Japan.

[72] Jun Rekimoto, M. G. (1993). The Information Cube: Using Transparency in 3d Information VisualizationInformation Cube. In *Proceedings of the Third Annual Workshop on Information Technologies & Systems (WITS'93)*, pages 125–132.

[73] Kennedy, R. S., Lane, N. E., Berbaum, K. S., and G.Lilienthal, M. (1993a). Simulator sickness questionnaire: An enhanced method for quantifying simulator sickness. *The International Journal of Aviation Psychology*, 3(3):203–220, DOI:10.1207/s15327108ijap0303_3.

[74] Kennedy, R. S., Lane, N. E., Berbaum, K. S., and Lilienthal, M. G. (1993b). Simulator Sickness Questionnaire: An Enhanced Method for Quantifying Simulator Sickness. *The International Journal of Aviation Psychology*, 3(3):203–220.

[75] Kipouros, T., Inselberg, A., Parks, G. T., and Savill, M. (2013). Parallel Coordinates in Computational Engineering Design. In *AIAA Multidisciplinary Design Optimization Specialists*, volume 1750.

[76] Kipouros, T., Jaeggi, D. M., Dawes, W N, P. G. T., Savill, M., and Clarkson, P. J. (2008a). Bi-objective design optimization for axial compressors using tabu search. *AIAA Journal*, 46(3):701–711.

[77] Kipouros, T., Jaeggi, D. M., Dawes, W N, P. G. T., Savill, M., and Clarkson, P. J. (2008b). Insight into high-quality aerodynamic design spaces through multi-objective optimization. *CMES: Computer Modeling in Engineering and Sciences*, 37(1):1–44.

[78] Kraus, M., Weiler, N., Oelke, D., Kehrer, J., Keim, D., and Fuchs, J. (2020). The impact of immersion on cluster identification tasks. *IEEE Transactions on Visualization and Computer Graphics*, 26:525–535.

[79] Kristensson, P. O., Dahlback, N., Anundi, D., Bjornstad, M., Gillberg, H., Haraldsson, J., Martensson, I., Nordvall, M., and Stahl, J. (2009). An evaluation of space time cube representation of spatiotemporal patterns. *IEEE Transactions on visualization and computer graphics*, 15(4):696–702.

[80] Kyritsis, M., Gulliver, S. R., Morar, S., and Stevens, R. (2013). Issues and Benefits of Using 3d Interfaces: Visual and Verbal Tasks. In *Proceedings of the Fifth International Conference on Management of Emergent Digital EcoSystems*, pages 241–245, NY, USA. ACM.

[81] Lam, H., Bertini, E., Isenberg, P., Plaisant, C., and Carpendale, S. (2012). Empirical Studies in Information Visualization: Seven Scenarios. *IEEE Transactions on Visualization and Computer Graphics*, 18(9):1520–1536.

[82] Larsson, T. (2008). Fast and tight fitting bounding spheres. In *SIGRAD 2008. The Annual SIGRAD Conference Special Theme: Interaction; November 27-28; 2008 Stockholm; Sweden*, pages 27–30. Linköping University Electronic Press.

[83] Latoschik, M. E. (2001). A Gesture Processing Framework for Multimodal Interaction in Virtual Reality. In *Proceedings of the 1st International Conference on Computer Graphics, Virtual Reality and Visualisation*, AFRIGRAPH '01, pages 95–100, New York, NY, USA. ACM. event-place: Camps Bay, Cape Town, South Africa.

[84] Leap Motion (Last accessed: Nov 2018). Leap motion. https://www.leapmotion.com/.

[85] Lecakes, G. D., Russell, M., Mandayam, S., Morris, J. A., and Schmalzel, J. L. (2009). Visualization of multiple sensor measurements in a VR environment for integrated systems health management in rocket engine tests. In *2009 IEEE Sensors Applications Symposium*, pages 132–136.

[86] LeCun, Y. and Cortes, C. (2010). MNIST handwritten digit database.

[87] Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al. (2017). Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4681–4690.

[88] Lewis, C., Rieman, J., and Blustein, A. J. (1993). Task-centered user interface design: A practical introduction. *https://hcibib.org/tcuid/*.

[89] Lin, L. and Jörg, S. (2016). Need a Hand?: How Appearance Affects the Virtual Hand Illusion. In *Proceedings of the ACM Symposium on Applied Perception*, SAP '16, pages 69–76, New York, NY, USA. ACM. event-place: Anaheim, California.

[90] Liu, R., Zhu, Y., Luo, Y., and Liu, X. (2009). Construction Urban Infrastructure Based on Core Techniques of Digital Photogrammetry and Remote Sensing. In *2009 International Forum on Information Technology and Applications*, volume 1, pages 536–539.

[91] Macchi, M., Roda, I., Negri, E., and Fumagalli, L. (2018). Exploring the role of digital twin for asset lifecycle management. *IFAC-PapersOnLine*, 51(11):790 – 795. 16th IFAC Symposium on Information Control Problems in Manufacturing INCOM 2018.

[92] Manders, C., Farbiz, F., Yin, T. K., Miaolong, Y., Chong, B., and Guan, C. G. (2008). Interacting with 3d Objects in a Virtual Environment Using an Intuitive Gesture System. In *Proceedings of The 7th ACM SIGGRAPH International Conference on Virtual-Reality Continuum and Its Applications in Industry*, VRCAI '08, pages 5:1–5:5, New York, NY, USA. ACM. event-place: Singapore.

[93] Mannino, M. and Abouzied, A. (2019). Is this real? generating synthetic data that looks real. In *Proceedings of the 32nd Annual ACM Symposium on User Interface Software and Technology*, UIST '19, page 549–561, New York, NY, USA. Association for Computing Machinery.

[94] Matejka, J. and Fitzmaurice, G. (2017). Same stats, different graphs: Generating datasets with varied appearance and identical statistics through simulated annealing. In *Proceedings of the 2017 CHI Conference on Human Factors in Computing Systems*, CHI '17, page 1290–1294, New York, NY, USA. Association for Computing Machinery.

[95] Meyburg, J. P. and Diesing, D. (2017). Teaching the Growth, Ripening, and Agglomeration of Nanostructures in Computer Experiments. *J. Chem. Educ.*, 94(9):1225–1231.

[96] Microsoft (Last accessed: Nov 2018). Xbox gaming platform. https://www.microsoft.com.

[97] Milgram, P., Takemura, H., Utsumi, A., and Kishino, F. (1995). Augmented reality: a class of displays on the reality-virtuality continuum.

[98] Mizell, D. W. (1994). Virtual reality and augmented reality in aircraft design and manufacturing. In *Proceedings of WESCON '94*, pages 91–.

[99] Murnane, M., Breitmeyer, M., Matuszek, C., and Engel, D. (2019). Virtual Reality and Photogrammetry for Improved Reproducibility of Human-Robot Interaction Studies. In *2019 IEEE Conference on Virtual Reality and 3D User Interfaces (VR)*, pages 1092–1093.

[100] Nalepa, G. J. (2018). *Modeling with Rules Using Semantic Knowledge Engineering*, volume 130 of *Intelligent Systems Reference Library*. Cham, Switzerland: Springer International Publishing, 1st edition.

[101] Nielsen, J. (1994a). Enhancing the Explanatory Power of Usability Heuristics. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '94, pages 152–158, New York, NY, USA. ACM.

[102] Nielsen, J. (1994b). Usability Inspection Methods. In *Conference Companion on Human Factors in Computing Systems*, CHI '94, pages 413–414, New York, NY, USA. ACM.

[103] Nishino, H., Utsumiya, K., Kuraoka, D., Yoshioka, K., and Korida, K. (1997). Interactive Two-handed Gesture Interface in 3D Virtual Environments. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '97, pages 1–8, New York, NY, USA. ACM. event-place: Lausanne, Switzerland.

[104] Nokuo, T. and Sumii, T. (2014). Head mounted display. US Patent App. 29/502,182.

[105] Oculus VR (Last accessed: Nov 2018b). Oculus Rift, Oculus Quest, Oculus Go. https://www.oculus.com.

[106] Oculus VR (Last accessed: Nov 2018c). Oculus Utilities for Unity 5. https://developer.oculus.com.

[107] Oculus VR ([Online]: Nov. 2018a). Oculus Rift. https://oculus.com.

[108] Poulton, E. and Freeman, P. (1966). Unwanted asymmetrical transfer effects with balanced experimental designs. *Psychological Bulletin*, 66(1):1.

[109] Prouzeau, A., Cordeil, M., Robin, C., Ens, B., Thomas, B., and Dwyer, T. (2019a). Scaptics and highlight-planes: immersive interaction techniques for finding occluded features in 3d scatterplots. In Cox, A. and Kostakos, V., editors, *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, United States of America. Association for Computing Machinery (ACM). International Conference on Human Factors in Computing Systems 2019, CHI 2019 ; Conference date: 04-05-2019 Through 09-05-2019.

[110] Prouzeau, A., Cordeil, M., Robin, C., Ens, B., Thomas, B. H., and Dwyer, T. (2019b). Scaptics and Highlight-Planes: Immersive Interaction Techniques for Finding Occluded Features in 3D Scatterplots. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, pages 1—12, New York, NY, USA. Association for Computing Machinery.

[111] Pukelsheim, F. (1993). *Optimal design of experiments*, volume 50. SIAM, 3600 Market Street, 6th Floor, Philadelphia, PA 19104-2688 USA.

[112] Rheinberg, F., Vollmeyer, R., and Engeser, S. (2003). Assessment of flow experiences. *Diagnosis of Motivation and Self-Concept (Tests and Trends N.F.2)*, pages 261—-279, eds. J. Stiensmeier–Pelster and F. Rheinberg (Göttingen: Hogrefe).

[113] Ribarsky, W., Ayers, E., Eble, J., and Mukherjea, S. (1994a). Glyphmaker: creating customized visualizations of complex data. *Computer*, 27(7):57–64.

[114] Ribarsky, W., Bolter, J., Op Den Bosch, A., and Van Teylingen, R. (1994b). Visualization and Analysis Using Virtual Reality. *Computer Graphics and Applications, IEEE*, 14:10–12.

[115] Ritchey, T. (2018). General morphological analysis as a basic scientific modelling method. *Technological Forecasting and Social Change*, 126:81–91.

[116] Ritter, J. (1990). An Efficient Bounding Sphere. In *Graphics Gems*, pages 301–303. Elsevier.

[117] Roberts, D. J., Garcia, A. S., Dodiya, J., Wolff, R., Fairchild, A. J., and Fernando, T. (2015). Collaborative telepresence workspaces for space operation and science. In *2015 IEEE Virtual Reality (VR)*, pages 275–276.

[118] Rokach, L. and Maimon, O. (2005). Clustering Methods. In Maimon, O. and Rokach, L., editors, *Data Mining and Knowledge Discovery Handbook*, pages 321–352. Springer US, Boston, MA.

[119] Rosenbaum, R., Bottleson, J., Liu, Z., and Hamann, B. (2011). Involve Me and I Will Understand!: Abstract Data Visualization in Immersive Environments. In *Proceedings of the 7th International Conference on Advances in Visual Computing - Volume Part I*, ISVC'11, pages 530–540, Berlin, Heidelberg. Springer-Verlag.

[120] Ruddle, R. A. (2004). The effect of environment characteristics and user interaction on levels of virtual environment sickness. *IEEE Virtual Reality 2004*, pages 141–285, DOI:10.1109/VR.2004.1310067.

[121] Russell, M., Lecakes, G. D., Mandayam, S., Morris, J. A., Turowski, M., and Schmalzel, J. L. (2009). Acquisition, interfacing and analysis of sensor measurements in a VR environment for integrated systems health management in rocket engine tests. In *2009 IEEE Sensors Applications Symposium*, pages 128–131.

[122] Russell Freeman ([Online]: Aug. 2020). Virtuality Continuum. https://en.wikipedia.org/wiki/File:Virtuality_Continuum_2.jpg.

[123] Sagardia, M., Hertkorn, K., Hulin, T., Wolff, R., Hummell, J., Dodiya, J., and Gerndt, A. (2013). An interactive virtual reality system for on-orbit servicing. In *2013 IEEE Virtual Reality (VR)*, pages 1–1.

[124] Savall, J., Borro, D., Gil, J. J., and Matey, L. (2002). Description of a haptic system for virtual maintainability in aeronautics. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 3, pages 2887–2892 vol.3.

[125] Schwind, V., Knierim, P., Haas, N., and Henze, N. (2019). Using Presence Questionnaires in Virtual Reality. In *Proceedings of the 2019 CHI Conference on Human Factors in Computing Systems*, CHI '19, pages 360:1–360:12, New York, NY, USA. ACM. event-place: Glasgow, Scotland Uk.

[126] Sedlmair, M., Tatu, A., Munzner, T., and Tory, M. (2012). A taxonomy of visual cluster separation factors. *Computer Graphics Forum*, 31(3pt4):1335–1344.

[127] See, Z. S., Santano, D., Sansom, M., Fong, C. H., and Thwaites, H. (2018). Tomb of a Sultan: A VR Digital Heritage Approach. In *2018 3rd Digital Heritage International Congress (DigitalHERITAGE) held jointly with 2018 24th International Conference on Virtual Systems Multimedia (VSMM 2018)*, pages 1–4.

[128] Seshadri, P., Duncan, A., Simpson, D., Thorne, G., and Parks, G. (2019a). Spatial flow-field approximation using few thermodynamic measurements part ii: Uncertainty assessments. *arXiv preprint arXiv:1908.02934*.

[129] Seshadri, P., Shahpar, S., Constantine, P., Parks, G., and Adams, M. (2018a). Turbomachinery active subspace performance maps. *Journal of Turbomachinery*, 140(4):DOI: 10.1115/GT2017–64528.

[130] Seshadri, P., Simpson, D., Thorne, G., Duncan, A., and Parks, G. (2019b). Spatial flow-field approximation using few thermodynamic measurements part i: Formulation and area averaging. *arXiv preprint arXiv:1908.03431*.

[131] Seshadri, P., Yuchi, S., Parks, G., and Shahpar, S. (2019c). Supporting multi-point fan design with dimension reduction. *under review, AIAA Journal of Propulsion and Power*.

[132] Seshadri, P., Yuchi, S., and Parks, G. T. (2018b). Dimension reduction via gaussian ridge functions. *SIAM/ASA J. of Uncertainty Quantification (accepted)*.

[133] Shakal, C., https://grabcad.com/chris.shakal (Last accessed: September 2019). Trent 900 turbofan model, https://grabcad.com/library/trent-900-turbofan-1. GrabCAD https://grabcad.com.

[134] Shefelbine, S., Clarkson, J., Farmer, R., and Eason, S. (2002). *Good Design Practice for Medical Devices and Equipment - Requirements Capture*. Cambridge, UK, University of Cambridge, Engineering Design Centre and University of Cambridge Institute for Manufacturing.

[135] Shneiderman, B. (1996). The eyes have it: a task by data type taxonomy for information visualizations. In *Proceedings 1996 IEEE Symposium on Visual Languages*, pages 336–343.

[136] Shneiderman, B. (2003a). Why not make interfaces better than 3D reality? *IEEE Computer Graphics and Applications*, 23(6):12–15.

[137] Shneiderman, B. (2003b). Why not make interfaces better than 3D reality? *IEEE Comput. Graph. Appl.*, 23(6):12–15, DOI:10.1109/MCG.2003.1242376.

[138] Shneiderman, B. and Plaisant, C. (2009). *Designing the User Interface: Strategies for Effective Human-Computer Interaction*. Addison-Wesley Publishing Company, USA, 5th edition.

[139] Sidenmark, L. and Gellersen, H. (2019a). Eye, Head and Torso Coordination During Gaze Shifts in Virtual Reality. *ACM Transactions on Computer-Human Interaction*, 27(1):4:1–4:40.

[140] Sidenmark, L. and Gellersen, H. (2019b). Eye&Head: Synergetic Eye and Head Movement for Gaze Pointing and Selection. *UIST*.

[141] Slambekova, D., Bailey, R., and Geigel, J. (2012). Gaze and Gesture Based Object Manipulation in Virtual Worlds. In *Proceedings of the 18th ACM Symposium on Virtual Reality Software and Technology*, VRST '12, pages 203–204, New York, NY, USA. ACM. event-place: Toronto, Ontario, Canada.

[142] Smoluchowski, M. v. (1906). Zur kinetischen Theorie der Brownschen Molekularbewegung und der Suspensionen. *Annalen der Physik*, 326(14):756–780.

[143] Song, C. G., Kwak, N. J., and Jeong, D. H. (2000). Developing an Efficient Technique of Selection and Manipulation in Immersive V.E. In *Proceedings of the ACM Symposium on Virtual Reality Software and Technology*, VRST '00, pages 142–146, New York, NY, USA. ACM. event-place: Seoul, Korea.

[144] Song, P., Goh, W. B., Hutama, W., Fu, C.-W., and Liu, X. (2012). A handle bar metaphor for virtual object manipulation with mid-air interaction. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, CHI '12, pages 1297–1306, Austin, Texas, USA. Association for Computing Machinery.

[145] Spurek, P., Winczowski, S., Tabor, J., Zamorski, M., Zięba, M., and Trzciński, T. (2020). Hypernetwork approach to generating point clouds. *arXiv:2003.00802 [cs]*. arXiv: 2003.00802.

[146] Stone, R. J., Panfilov, P. B., and Shukshunov, V. E. (2011). Evolution of aerospace simulation: From immersive Virtual Reality to serious games. In *Proceedings of 5th International Conference on Recent Advances in Space Technologies - RAST2011*, pages 655–662.

[147] Streit, M., Ecker, R. C., Österreicher, K., Steiner, G. E., Bischof, H., Bangert, C., Kopp, T., and Rogojanu, R. (2006). 3d parallel coordinate systems—A new data visualization method in the context of microscopy-based multicolor tissue cytometry. *Cytometry Part A*, 69A(7):601–611.

[148] Stump, G. M., Yukish, M., Simpson, T. W., and O'Hara, J. J. (2004). Trade space exploration of satellite datasets using a design by shopping paradigm. In *2004 IEEE Aerospace Conference Proceedings (IEEE Cat. No.04TH8720)*, volume 6, pages 3885–3895 Vol.6.

[149] Sutherland, I. E. (1968). A Head-mounted Three Dimensional Display. *AFIPS '68 (Fall, part I) Proceedings of the December 9-11, 1968, fall joint computer conference, part I*, pages 757–764, DOI:10.1145/1476589.1476686.

[150] Tadeja, S. K., Kipouros, T., and Kristensson, P. O. (2019a). Exploring Parallel Coordinates in Virtual Reality. In *Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems (CHI'19)*, Glasgow, Scotland UK.

[151] Tadeja, S. K., Kipouros, T., and Kristensson, P. O. (2020a). IPCP: Immersive Parallel Coordinates Plots for Engineering Design Processes. In *Proceedings of AIAA SciTech Forum and Exposition*, Orlando, Florida.

[152] Tadeja, S. K., Langdon, P., and Kristensson, P. O. (2021a). Supporting Iterative Virtual Reality Analytics Design and Evaluation by Systematic Generation of Surrogate Clustered Datasets. In *IEEE International Symposium on Mixed and Augmented Reality (ISMAR 2021) (forthcoming)*.

[153] Tadeja, S. K., Lu, Y., Rydlewicz, M., Rydlewicz, W., Bubas, T., and Kristensson, P. O. (2021b). Exploring Gestural Input for Engineering Surveys of Real-life Structures in Virtual Reality using Photogrammetric 3D Models. *Multimedia Tools and Applications*.

[154] Tadeja, S. K., Lu, Y., Seshadri, P., and Kristensson, P. O. (2020b). Digital Twin Assessments in Virtual Reality: An Explorational Study with Aeroengines. In *The 41st IEEE Aerospace Conference*.

[155] Tadeja, S. K., Rydlewicz, W., Lu, Y., Kristensson, P. O., and Bubas, T. Rydlewicz, M. (2019b). PhotoTwinVR: An Immersive System for Manipulation, Inspection and Dimension Measurements of the 3D Photogrammetric Models of Real-Life Structures in VR. https://arxiv.org/abs/1911.09958.

[156] Tadeja, S. K., Seshadri, P., and Kristensson, P. O. (2019c). Exploring Aerospace Design in Virtual Reality with Dimension Reduction. In *Proceedings of AIAA SciTech Forum and Exposition*, 7 - 11 Jan 2019 San Diego.

[157] Tadeja, S. K., Seshadri, P., and Kristensson, P. O. (2020c). AeroVR: Immersive Visualization System for Aerospace Design and Digital Twinning in Virtual Reality. *The Aeronautical Journal*, pages 1615–1635.

[158] Tadeja, S. K., Spurek, P., Jacques, J., and Kristensson, P. O. (2020d). Practical Approaches for Providing Quality Content for Virtual Reality: A Short Synthesis of Existing Methods. *(draft manuscript)*.

[159] Tecchia, F., Avveduto, G., Brondi, R., Carrozzino, M., Bergamasco, M., and Alem, L. (2014). I'm in VR!: Using Your Own Hands in a Fully Immersive MR System. In *Proceedings of the 20th ACM Symposium on Virtual Reality Software and Technology*, VRST '14, pages 73–76, New York, NY, USA. ACM. event-place: Edinburgh, Scotland.

[160] Theodoridis, Y., Silva, J. R. O., and Nascimento, M. A. (1999). On the generation of spatiotemporal datasets. In *Proceedings of the 6th International Symposium on Advances in Spatial Databases*, SSD '99, page 147–164, Berlin, Heidelberg. Springer-Verlag.

[161] Thomas, J. J. and Cook, K. A. (2006). A visual analytics agenda. *IEEE Comput. Graph. Appl.*, 26(1):10–13.

[162] Thompson, M. M. (1966). Manual of photogrammetry. *Geoscience Abstracts*.

[163] Tory, M. and Moller, T. (2004). Human factors in visualization research. *IEEE transactions on visualization and computer graphics*, 10(1):72–84.

[164] Tukey, J. W. and Tukey, P. A. (1985). Computer Graphics and Exploratory Data Analysis: An Introduction. In *Proc. the Sixth Annual Conference and Exposition: Computer Graphics '85, Vol. III, Technical Sessions*, pages 773–785. Nat. Computer Graphics Association.

[165] Unity Game Engine (Last accessed: Oct 2019). Unity. https://unity.com/.

[166] Unity3D Game Engine (Last accessed: Nov 2018). Unity VR Samples pack. https://assetstore.unity.com/packages/essentials/tutorial-projects/vr-samples-51519.

[167] Vajak, D. and Livada, Č. (2017). Combining photogrammetry, 3D modeling and real time information gathering for highly immersive VR experience. In *2017 Zooming Innovation in Consumer Electronics International Conference (ZINC)*, pages 82–85.

[168] van der Maaten, L. and Hinton, G. (2008). Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9:2579–2605.

[169] Van Wijk, J. J. (2006). Views on visualization. *IEEE transactions on visualization and computer graphics*, 12(4):421–432.

[170] Wagner, M., Blumenstein, K., Rind, A., Seidl, M., Schmiedl, G., Lammarsch, T., and Aigner, W. (2016). Native Cross-Platform Visualization: A Proof of Concept Based on the Unity3d Game Engine. *20th International Conference Information Visualisation.*, pages 39–44, DOI:10.1109/IV.2016.35.

[171] Wagner Filho, J. A., Freitas, C., and Nedel, L. (2018). Virtualdesk: A comfortable and efficient immersive information visualization approach. *Computer Graphics Forum*, 37(3):415–426.

[172] Ward, R., Choudhary, R., Heo, Y., and Aston, J. (2019). A data-centric bottom-up model for generation of stochastic internal load profiles based on space-use type. *Journal of Building Performance Simulation*, pages 1–17.

[173] Wegenkittl, R., Loffelmann, H., and Groller, E. (1997). Visualizing the behaviour of higher dimensional dynamical systems. In *Proceedings. Visualization '97 (Cat. No. 97CB36155)*, pages 119–125.

[174] Wegman, E. J. and Luo, Q. (1997). High Dimensional Clustering Using Parallel Coordinates and the Grand Tour. In Klar, R. and Opitz, O., editors, *Classification and Knowledge Organization*, Studies in Classification, Data Analysis, and Knowledge Organization, pages 93–101. Springer Berlin Heidelberg.

[175] Weiss, G. H. (1994). *Aspects and Applications of the Random Walk*. Elsevier Science Ltd, Amsterdam The Netherlands ; New York.

[176] Wiener, N. (1923). Differential-Space. *Journal of Mathematics and Physics*, 2(1-4):131–174.

[177] Wijk, J. J. v. (2013). Evaluation: A Challenge for Visual Analytics. *IEEE Computer Society*, 46(7):56–60.

[178] Wilkinson, L., Anand, A., and Grossman, R. (2005). Graph-Theoretic Scagnostics. In *Proceedings of the Proceedings of the 2005 IEEE Symposium on Information Visualization*, INFOVIS '05, Washington, DC, USA. IEEE Computer Society.

[179] Wilkinson, L., Anand, A., and Grossman, R. (2006). High-Dimensional Visual Analytics: Interactive Exploration Guided by Pairwise Views of Point Distributions. *IEEE Trans. Vis. Comput. Graphics*, 12(6):1363–1372.

[180] Wiss, U., Carr, D., and Jonsson, H. (1998). Evaluating three-dimensional information visualization designs: a case study of three designs. In *Proceedings. 1998 IEEE Conference on Information Visualization. An International Conference on Computer Visualization and Graphics (Cat. No.98TB100246)*, pages 137–144.

[181] Wright, J., Hartman, F., and Cooper, B. (2001). Immersive environment technologies for planetary exploration. In *Proceedings IEEE Virtual Reality 2001*, pages 183–190.

[182] Wu, Z., Song, S., Khosla, A., Yu, F., Zhang, L., Tang, X., and Xiao, J. (2015). 3d shapenets: A deep representation for volumetric shapes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1912–1920.

[183] Xu, D. and Tian, Y. (2015). A Comprehensive Survey of Clustering Algorithms. *Ann. Data. Sci.*, 2(2):165–193.

[184] Yang, Y., Cordeil, M., Beyer, J., Dwyer, T., Marriott, K., and Pfister, H. (2020). Embodied navigation in immersive abstract data visualization: Is overview+detail or zooming better for 3d scatterplots?

[185] Zhang, J., Chen, Y., Hashizume, S., Muramatsu, N., Omomo, K., Iwasaki, R., Wataru, K., and Ochiai, Y. (2018). EXController: Enhancing Interaction Capability for VR Handheld Controllers Using Real-time Vision Sensing. In *Proceedings of the 24th ACM Symposium on Virtual Reality Software and Technology*, VRST '18, pages 87:1–87:2, New York, NY, USA. ACM. event-place: Tokyo, Japan.

Appendix A

Surrogate Dataset Generation Method: Pseudo-Code Procedures

As mentioned in Chapter 6, the dataset's generative process (see `GenerateCluster()`) can be split into these four steps :

(1) determine the position of the central point of each cluster in 3D space
(see `DrawBrownianTrail()` and `DrawFromPoisson()`);

(2) determine the size of each individual cluster
(see `DrawSizeFromPower()`);

(3) generate samples of data around each clusters' central points
(see `Generate3DPointCloud()`);

(4) colour-code all individual clusters
(see `AssignColors()`).

This Appendix presents the pseudo-code for both the dataset's generation process and clusters' colour-coding procedures.

```
Require:
    number of clusters $N$, set of $K$ distinguished colours
Ensure:
    $N \geq 0, K > 0$
$clusters \leftarrow$ GetEmptyList($void$)
$D_{n,n} \leftarrow$ GetEmptyMatrix($void$)
$i \leftarrow 1$
while $i \leq N$ do
    $cluster \leftarrow$ GenerateCluster($N, clusters$)
    AppendCluster($cluster, clusters$)
    $i \leftarrow i + 1$
Assigncolors($clusters, K$)

procedure GenerateCluster($N, clusters$)
    $brownian \leftarrow$ DrawBrownianTrail($2N$)
    while $True$ do
        $s \leftarrow$ DrawFromPoisson($void$)
        if $s \leq N$ then
                                        ▷ Append current trail with $s - N$ samples.
            $brownian \leftarrow$ AppendBrownianTrail($N + s$)
            $N = N + s$
        $P \leftarrow brownian[s]$
        if $P$ was not drawn before then
            $size \leftarrow$ DrawSizeFromPower($void$)
            $cluster \leftarrow$ Generate3DPointCloud($P, size$)
            $S \leftarrow$ SetBoundingSphere($cluster, r, O$)
            if $S$ overlap acceptable with others then
                ▷ Updated the matrix $D_{n,n}$ with distances calculated whilst evaluating $if$ statement.
                UpdatedMatrix($D_{n,n}$)
                return $cluster$

procedure Generate3DPointCloud($P, n$)
                                        ▷ Generate cluster of size $n$ using $\mathcal{N}(\sigma^2, \mu)$.
    $s \leftarrow$ SetSigma($void$)
    $points \leftarrow$ GetEmptyList($void$)
    while $i \leq n$ do
        $p_x \leftarrow$ GetCoordFromNormDist($\sigma^2 = s, \mu = P_x$)
        $p_y \leftarrow$ GetCoordFromNormDist($\sigma^2 = s, \mu = P_y$)
        $p_z \leftarrow$ GetCoordFromNormDist($\sigma^2 = s, \mu = P_z$)
        AppendPoint($p, points$)
        $i \leftarrow i + 1$
    return $points$

procedure SetBoundingSphere($C, r, O$)
                                        ▷ Set bounding sphere surrounding entire cluster $C$.
    $S \leftarrow$ SetSphere($C, r, O$)
    return $S$
```

Algorithm 1 An algorithm describing in pseudo-code the dataset generation method.

```
Require:
    list of $clusters$
    cet of $K$ indexed $colors$
Ensure:
    $K > 0$
procedure AssignColors($clusters, K$)
    $N$ = len($clusters$)
    if $N \leq K$ then                                    ▹ If we have less clusters than colors.
        colorClusters($K, clusters$)
        exit($SUCCESS$)
    $picked \leftarrow$ GetEmptyList($N$)
    $picked[0] = colors[0]$                                          ▹ Set color of first cluster.
    $i \leftarrow 0$
    while $i < N$ do
        $A \leftarrow$ GetEmptyList($K$)
        $j \leftarrow 0$
        while $j < i$ do
            if $w_{i,j} \neq 0$ then
                $A[i]$ = GetCoeff($colors, i, j, K, A$)
            $j \leftarrow j + 1$
        $picked[i] = colors$[MapToIndex($A[i]$)]
        $i \leftarrow i + 1$
    colorClusters($picked, clusters$)

procedure GetCoeff($colors, i, j, K, A$)
    $k \leftarrow 0$
    while $k < K$ do
        $d = |colors[i] - colors[j]|$
        if $K$ is even then
            $A[k] = w_{i,j}^{-2}(K - \mathrm{min}(d, K - d))$
        else
            $A[k] = w_{i,j}^{-2}(K - \mathrm{min}(d, K + 1 - d))$
        $k \leftarrow k + 1$
    return argmin($A$)
```

Algorithm 2 An algorithm describing in pseudo-code the color-coding procedure.